CURSO DE TECNOLOGIA EM DESENVOLVIMENTO DE SOFTWARE

NILCIMAR DOS SANTOS SOUZA

DESENVOLVIMENTO DO ESPAÇO VIRTUAL DE APRENDIZAGEM – EVA UTILIZANDO O SISTEMA GERENCIADOR DE CONTEÚDO XOOPS

Campos dos Goytacazes/RJ
2008

NILCIMAR DOS SANTOS SOUZA

DESENVOLVIMENTO DO ESPAÇO VIRTUAL DE APRENDIZAGEM – EVA UTILIZANDO O SISTEMA GERENCIADOR DE CONTEÚDO XOOPS

Monografia apresentada ao curso superior de Desenvolvimento de Software do Centro Federal de Educação Tecnológica, como parte das exigências para a obtenção do título de tecnólogo em Desenvolvimento de Software.

Orientador: Prof. D.Sc Ernesto Macedo Reis

Campos do Goytacazes – RJ
Março – 2008

NILCIMAR DOS SANTOS SOUZA

DESENVOLVIMENTO DO ESPAÇO VIRTUAL DE APRENDIZAGEM – EVA UTILIZANDO O SISTEMA GERENCIADOR DE CONTEÚDO XOOPS

Monografia apresentada ao curso superior de Desenvolvimento de Software do Centro Federal de Educação Tecnológica, como parte das exigências para a obtenção do título de tecnólogo em Desenvolvimento de Software.

Aprovada em 19 de março de 2008

Banca Avaliadora:

____
Prof. D.Sc. Ernesto Macedo Reis – CEFET-Campos (orientador)

____
Profª. D.Sc. Marília Paixão Linhares – UENF

____
Prof. M.Sc. Maurício José Viana Amorim – CEFET-Campos

Campos do Goytacazes – RJ
Março – 2008

AGRADECIMENTOS

Agradeço inicialmente a Ernesto Macedo Reis, pelo imenso incentivo e motivação ao longo da orientação de todo o projeto e em especial desta monografia. Ao longo do projeto me mostrou a possibilidade de unir na Informática Educativa duas áreas que a princípio pareciam imiscíveis: a Química e a Computação. Evidentemente sem sua contribuição ao longo destes anos este trabalho não seria possível.

No mesmo patamar de importância, agradeço à Marília Paixão Linhares pela participação na realização deste trabalho e dedicação no encaminhamento do projeto de pesquisa, além da aceitar compor da banca de avaliação.

Aos meus pais por tudo.

À todos os colegas das 3 turmas de tecnólogo que tive a oportunidade de participar, em especial a última, na qual cursei 5 dos 6 semestres.

Ao CEFET-Campos por estar sempre preocupada em oferecer os melhores recursos aos estudantes.

À todos os colegas do grupo de pesquisa. Do CEFET: primeiramente a David Vasconcelos, por ter confiado a mim a oportunidade de participar deste projeto, posteriormente a Marília Dutra e Lucas Sepulvida por compartilhar os momentos de desenvolvimento e demais decisões tecnológicas. Da UENF: Maria Helena, João Paulo, Renata Lacerda, Ronaldo Bastos, Vanessa Oliveira e Cassiana Hygino, sendo esta parceira de viagens, ora de ida da casa ao CEFET, ora de volta da UENF a casa.

Ao professor Maurício José Viana Amorim por aceitar compor a banca de avaliação.

Resumo: Descrevemos neste trabalho a reformulação de um ambiente virtual de aprendizagem denominado Espaço Virtual de Aprendizagem (EVA), a favor de uma solução baseada na utilização do Sistema Gerenciador de Conteúdo XOOPS, cujas características devem atender a necessidade de aplicação do modelo de Aprendizagem Baseada em Casos na Internet na área do Ensino de Ciências. O Sistema Gerenciador de Conteúdo XOOPS foi criado por desenvolvedores do Japão e China, no entanto, por ter sido concebido como software livre, é desenvolvido colaborativamente por centenas de pessoas em todo o mundo atualmente. Sua utilização na reformulação do sistema EVA é propícia neste momento em que cresce a necessidade de explorar as oportunidades de flexibilizar a criação e gestão de conteúdo com tecnologias deste porte nos sistemas de informação e comunicação. O sistema foi testado, com caráter de uma versão beta, junto a um grupo de estudantes de Ensino Médio do CEFET-Campos durante um bimestre letivo.

Palavras-chave: Ambiente Virtual de Aprendizagem. Sistema Gerenciador de Conteúdo. XOOPS

Abstract: We describe in this work the reformulation of a virtual learning environment called Virtual Learning Space (EVA – Espaço Virtual de Aprendizagem), in favor of a solution based on the use of the Content Management System XOOPS, whose characteristics must attend to the need for application of the Case Based Learning Model on the Internet in the Science Teaching area. The Content Management System XOOPS was created by developers in Japan and China, however, because it was conceived as free software, is developed collaboratively by hundreds of people in the world nowadays. His use in the recasting of the EVA system is propitious in this moment which grows the need to explore opportunities to ease the creation and management of content with technologies of this size in information and communication systems. The system was tested, as beta version, among a group of high school students from CEFET-Campos during a two-month school period.

Keywords: Virtual Learning Environment. Content Management System. XOOPS

SUMÁRIO

RELAÇÃO DE ANEXOS

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE QUADROS

LISTA DE SIGLAS

ABC – Aprendizagem Baseada em Casos
AVA – Ambientes Virtuais de Aprendizagem
AVEC – Ambiente Virtual para o Ensino de Ciências
CAPTCHA – Completely Automated Public Turing test to tell Computers and Humans Apart
CMS – Content Management System
CSS – Cascading Style Sheets
DED – Diagrama de Estrutura de Dados
DER – Diagrama de Entidade Relacionamento
EDUCOM – Educação com Computadores
EVA – Espaço Virtaul de Aprendizagem
GPL – General Public License
HTML – HyperText Markup Language
InfoEduc – Informática Educativa
PHP – Page Hypertext Preprocessor
PROINFO – Programa Nacional de Informática na Educação
PRONINFE – Programa Nacional de Informática
RSS – Really Simple Syndication
SBIE – Simpósio Brasileiro de Informática Educativo
SBPC – Simpósio Brasileiro para o Progresso da Ciências
SECTEC – Secretaria de Ciência e Tecnologia do Piauí
SGC – Sistema Gerenciador de Conteúdo
SQL – Structured Query Language
TCDF – Tribunal de Contas do Distrito Federal
TIC – Tecnologias da Informação e Comunicação
URL – Uniform Resource Locator
WYSIWYG – What You See Is What You Get
XML – eXtensible Markup Language
XOOPS* – eXtensible Object Oriented Portal System

---
* Pronuncia-se “zú’ps”

LISTA DE TERMOS TÉCNICOS[1]

Backup – Refere-se à cópia de dados de um dispositivo para o outro com o objetivo de recuperá-los posteriormente, caso haja algum problema com os dados originais.

Bug – Erro no funcionamento comum de um software.

Framework – Estrutura de software definida que dá suporte para que outro projeto de software possa ser organizado e desenvolvido, podendo incluir programas de suporte, bibliotecas de código, linguagens de script.

Restore – Consiste em um conjunto de atividades que visam restabelecer os dados decorrentes de danos ao banco de dados.

Templates – Instrumentos utilizados para separar a parte de apresentação da parte lógica de um site visando a organização do código.

---

[1] Devido a grande quantidade de termos técnicos e termos em inglês presente neste trabalho, optamos por não apresentá-los em itálico.

Capítulo I

INTRODUÇÃO

Neste trabalho aborda-se o desenvolvimento de uma tecnologia de porte educacional, mais especificamente, um ambiente virtual de aprendizagem, que deve atender interesses de ensino e de pesquisa. O ambiente denominado Espaço Virtual de Aprendizagem foi construído utilizando-se uma ferramenta de gerência de conteúdo denominado XOOPS (acrônimo do inglês eXtensible Object Oriented Portal System).

Esta ferramenta constitui-se um gerenciador de conteúdo de código livre criado por desenvolvedores japoneses e chineses e desenvolvido cooperativamente por comunidades em todo o mundo. O XOOPS se mostrou adequado ao momento do desenvolvimento do Espaço Virtual de Aprendizagem, quando a versão anterior precisou ser reformulada visando atender interesses identificados na área do Ensino de Ciências (Reis, 2008).

As pesquisas sobre a utilização de Tecnologias de Informação e Comunicação (TIC) na educação teve seu marco inicial no Brasil, em um seminário ocorrido em 1971, no qual discutiu-se o uso de computadores no ensino de Física. Vários projetos foram criados com o objetivo de integrar o computador no processo de ensino e aprendizagem, sendo os mais abrangentes o projeto EDUCOM (Educação com Computadores – UFRJ, UNICAMP, UFRGS, UFPE, UFMG), em 1984/1992, o projeto PRONINFE (Programa Nacional de Informática), em 1989/1997 e o PROINFO (Programa Nacional de Informática na Educação), em 1997/atual (Moraes, 1997).

Os projetos envolvendo o uso das TIC no ensino de Ciência são direcionados atualmente, em grande parte, à utilização de plataformas de ensino. As plataformas se comportam como lançadoras de cursos e, portanto eximem-se da necessidade de uma teoria educacional orientadora, o que torna "*os ambientes virtuais de aprendizagem uma das tecnologias de informação e comunicação mais valorizadas no âmbito da construção e transformação de conhecimento*" (Reis, 2008).

Valente (2002) afirma que um sistema computacional com finalidades educacionais não pode ser utilizado sem considerar o seu contexto pedagógico de uso. Assim a construção e transformação de conhecimento não se caracterizam pelo simples acesso à Internet e as informação nela disponibilizadas. Sobretudo, é necessário transformar informação em conhecimento, e aí a escola se mostra como o ambiente natural e propício a esta transformação.

O desenvolvimento de ambientes virtuais de aprendizagens referenciados por sistema didático foi preterido pelas plataformas devido a complexidade computacional de desenvolver um ambiente virtual de aprendizagem, já que as plataformas podem ser utilizadas de acordo com diferentes sistemas gerenciais (concessões, comerciais etc).

Pretende-se com este trabalho investir na utilização do XOOPS como uma ferramenta informática atuante na integração de componentes necessárias para criar, estruturar, gerenciar, administrar e publicar ambientes virtuais de aprendizagem sem a necessidade de longos períodos de programação. Na literatura não há registros da utilização do XOOPS no desenvolvimento de ambientes virtuais de aprendizagem.

1.1 – Justificativa

O Espaço Virtual de Aprendizagem EVA[2] surge da evolução tecnológica de outro sistema (Reis, 2001), inicialmente vinculado à plataforma proprietária e posteriormente planejado e escrito em código livre.

No período compreendido entre os anos de 2002 e 2004 deu-se ênfase a reestruturação do ambiente baseado no princípio de independência da plataforma comercial. As novas versões projetadas no período AVEMA/2002 e AVEC – 2003/2004 passaram a rodar em servidores no CEFET-Campos (2002/2003) e UENF (2003/2004). As versões desenvolvidas, (Paula e da Hora, 2002) e (Azevedo, 2003), não foram validadas em cursos ou oficinas, apenas testadas com grupos de estudantes, professores e licenciandos nas áreas de Ciências Física e Química, com objetivos de testes das interfaces. O desempenho do sistema escrito em código livre, em um primeiro momento, mostrou diversos pontos de estrangulamento e o sistema, apesar de mais acessível e leve, passou a apontar problemas de interface, estrutura e documentação:

Interface

- Ergonomia das telas de apresentação e
- Navegabilidade entre as telas de apresentação.

Estruturais

- Impossibilidade de separar os alunos e professores em grupos;
- Erro no tratamento de datas;
- Impossibilidade de cadastrar visitantes;

[2] Disponível à visitação em www.uenf.t5.com.br

- Falta de ferramentas wysiwyg;
- Falta de ferramentas para repositórios de arquivos;
- Flexibilidade nula para adição de novos recursos ou melhorias;
- Inexistência de distinção entre os atores e
- Despadronização.

Documentação

- Padrões de projeto e
- Estrutura de banco de dados.

Todas as correções extrapolam a esfera da correção de bugs de código, e atingem abordagens sistêmicas, como por exemplo, a impossibilidade de atribuir permissões a diferentes tipos de usuários e associá-los à grupos de estudos. As falhas de caráter sistêmicas aliadas a baixa flexibilidade de adaptação do ambiente, justificaram a decisão de desenvolver um novo ambiente virtual de aprendizagem baseado no modelo conceitual já existente, mas utilizando a orientação dos sistemas gerenciadores de conteúdo em seu desenvolvimento.

Nesse estágio do projeto foi necessária a inserção de mais um bolsista visando o processo de modelagem e desenvolvimento do sistema. Neste sentido, é preciso explicitar as motivações da decisão de participar da equipe de desenvolvimento do ambiente virtual de aprendizagem EVA.

O trabalho insere-se na área da Informática Educativa, em uma das linhas de pesquisa que mais tem se difundido nos últimos dois anos, ambientes virtuais de aprendizagem. Enquanto atividade proposta em projeto de Iniciação Científica se justificativa a partir da formação como Tecnólogo em Informática – modalidade Desenvolvimento de Software.

Ao ser aprovado como bolsista foi possível, mediante contato permanente com o grupo de pesquisa, estudar os princípios de engenharia de software necessários ao desenvolvimento e implementação do sistema didático EVA. A inserção em atividades com o grupo de pesquisa possibilitou a apropriação de conhecimentos da área educacional.

Esta área de pesquisa incorpora outras motivações, que evoluem a partir de um curso técnico em Química (2002/2004), pós-técnico em Especialização em Fluidos de Perfuração e Completação de Poços de Petróleo (2004) e Licenciatura em Química (2005/atual).

As reuniões do projeto de pesquisa possibilitaram a identificar das relações entre o ensino de Ciências e as tecnologias, em especial as tecnologias educacionais. A possibilidade

de aliar as áreas de formação e interesse, o Ensino de Ciências e o Desenvolvimento de Software foi o fator fundamental na opção deste projeto em detrimento de outros.

A motivação ao longo do projeto foi se ampliando a partir das ações da pesquisa, em especial as relacionadas a este projeto de Iniciação Científica, já que proporcionaram a aplicação de conhecimentos construídos durante o Curso de Tecnologia em Desenvolvimento de Software, que puderam ser compreendidos como compatíveis com a área da Informática Educativa.

A área da Informática Educativa é uma das que mais tem crescido no Brasil e no mundo, apresentando-se como perspectiva acadêmica e empresarial. O mercado do software educativo e dos ambientes virtuais de aprendizagem está sendo apontado como a tendência atual de treinamento, aprendizagem e formação continuada no setor empresarial. Ele surgiu a partir das necessidades de empresas com o treinamento de seus funcionários. Porém, devido ao descaso com essa formação nas graduações a escassez de mão de obra qualificada não permite um avanço maior. Somente na pós-graduação têm-se notícia dessa formação que é técnica e humanística.

A importância desta área de pesquisa nos meios empresariais e acadêmicos foi comprovado nos eventos dos quais participamos durante o projeto, apresentando trabalhos, SBIE–2006/(Pamplona *et al*, 2006), Virtual Educa–2007/(Souza *et al*, 2007a) e SBPC–2007 (Souza *et al*, 2007b).

Pode-se apresentar então a questão da monografia: como agir para reestruturar o sistema que então denominaremos EVA de maneira a colocá-lo em funcionamento?

1.2 – Objetivo

O objetivo deste trabalho é a reformulação do sistema didático AVEC a favor de uma solução baseada na utilização do sistema gerenciador de conteúdo XOOPS.

São objetivos específicos:

1. Empregar a arquitetura modular ao sistema didático EVA e verificar seu desempenho.

2. Verificar a adequação dos padrões da engenharia de software ao desenvolvimento de um software educativo.

3. Desenvolver atividades de documentação ao desenvolvimento do sistema com finalidade de registro.

1.3 – Estrutura da monografia

Nesta monografia estão descritas todas as etapas percorridas para estruturação e disponibilização do ambiente. Para tanto, o texto está organizado em mais 4 capítulos além da presente introdução. O capítulo dois trata as teorias que orientam a utilização de ambientes virtuais de aprendizagem com suporte na Internet no ensino de ciências. Também é apresentado o referencial teórico orientador da proposta de desenvolvimento.

No capítulo três são apresentadas as etapas inerentes ao processo de desenvolvimento do sistema didático EVA.

No capítulo quatro são apresentados os resultados obtidos durante a utilização do sistema junto a grupo de estudantes do Ensino Médio, além de uma breve análise.

No quinto capítulo são feitas algumas considerações finais sobre o trabalho.

Capítulo II

REFERENCIAL TEÓRICO

O referencial teórico que orienta o desenvolvimento do Espaço Virtual de Aprendizagem sob o aspecto metodológico está baseado nos padrões e teorias da engenharia de software (Pressman, 2007 e Sommerville, 2007).

Quanto ao aspecto educacional a orientação é feita a partir das teorias relacionadas à Aprendizagem Baseada em Casos – ABC (Savery e Duffy, 1995), que organizam e ajudam a estruturar o sistema didático EVA.

A primeira seção trata da utilização de ambientes virtuais de aprendizagem com suporte na Internet no ensino de ciências e o referencial teórico que guiou o desenvolvimento do desenho instrucional do sistema. Na seção seguinte é apresentado o referencial teórico metodológico orientador da proposta de desenvolvimento do sistema: a utilização de Sistema Gerenciador de Conteúdo como plataforma de desenvolvimento do sistema, aportados em conceitos da engenharia de software e UML.

2.1 - Ambientes Virtuais Aprendizagem

> *Ambiente [ambi] – ao redor de – [ente] – noção de ação, qualidade, estado – nos conduz à idéia de que tal palavra remete a um lugar, espaço ou recinto envolvente e que por isso pode, ecologicamente e dinamicamente, propiciar as inter-relações entre organismos e seu meio circundante.* (Pequeno *et al*, 2002).

Uma extensão do significado de ambiente é ambiente de aprendizagem, que conforme a definição anterior é um local onde se propiciaria inter-relações entre alunos, entre professores e entre alunos e professores a fim de corroborar no processo de construção de conhecimento.

Para Reis e Linhares (2005b),

> *Um cenário que oriente a utilização das tecnologias nas aulas de Ciências deve compreender as relações sociais como fator de construção e apropriação do conhecimento e considerar o professor como mediador do processo pedagógico de ensino, além de entender os saberes dos educandos.*

Valente (2002) destaca o caráter educacional das tecnologias de informação e comunicação considerando que elas facilitam a transmissão de instruções, porém não mudarão a natureza da educação sem que mudanças fundamentais nas concepções e métodos

de ensino e aprendizagem se realizem. Portanto, a informatização do ensino tradicional tal qual o próprio ensino tradicional "*são baseados na transmissão de conhecimento, havendo apenas a transferência do professor para o computador da responsabilidade de ser o proprietário do saber*" (Valente, *op. cit.*).

Neste contexto predomina a "*educação bancária*" que entende o aluno como um recipiente a ser preenchido com informações do professor ou do computador. Este formato de educação é criticado por Freire (1970) em todos os seus contextos e aprendizagem.

Na definição de Valente (*op. cit.*) percebe-se a importância de não se conduzir a uma informatização do ensino tradicional ao desenvolver ambientes virtuais de aprendizagem, pois a simples composição de ambiente virtual de aprendizagem torna-se inócua se não apoiada por uma proposta pedagógica consistente. Este é o principal ponto que diferencia um ambiente virtual de aprendizagem das plataformas de ensino eletrônico e a distância.

Um ambiente virtual de aprendizagem se forma quando há um modelo pedagógico empregado voltado unicamente para ser meio facilitador e colaborador no processo de aprendizagem. No entanto, algumas vezes, os sistemas informáticos baseiam-se em propiciar o lançamento de cursos sem uma identidade com algum modelo de ensino. Esses sistemas informáticos são conhecidos como plataformas e se comportam como lançadoras de cursos.

Para Dillenbourg (2003):

> *Uma característica particular de ambientes virtuais de aprendizagem, em função das particularidades da Internet, é que os estudantes não estão restritos a consultar as informações da Rede, eles se tornam produtores da informação, participantes do jogo de aprender.*

No contexto, da cooperação, da aprendizagem e da autonomia é que se consolida o modelo pedagógico com características construtivista do sistema didático EVA. (Reis, 2008).

A concepção pedagógica de um ambiente construtivista de aprendizagem (Struchiner *et. al*., 1998) é mantida no sistema didático EVA desde o sistema AVEC (Reis, 2001) com alterações e tem como objetivo ampliar e potencializar a participação do aprendiz no seu próprio processo de aprendizagem, visando a troca de experiências entre os participantes, embasado em elementos teóricos da Aprendizagem Baseada em Casos - ABC (Savery & Duffy, 1995 e Schank & Clery, 1995).

2.2 – Sistemas gerenciadores de conteúdo

> *Para mejorar las crecientes expectativas del software educativo, diversos autores han señalado la necesidad de modificar el proceso de desarrollo, para encaminarlo hacia una mayor reutilización composicional, frente al paradigma actualmente predominante que es el generativo.* (Roselló *et al*, 2002).

Esta necessidade observada surge no contexto em que diversas ferramentas de apoio ao desenvolvimento de sistemas de informação e comunicação são produzidas a cada ano e logo absorvidas por organizações empresariais, no sentido de flexibilizar a gestão de seus portais Web:

> *A maioria das organizações usam seus site Web para comunicação com seus colaboradores, publicidade, publicação de notícias, eventos, etc. Nestas situações, há um problema em comum que é o gerenciamento de todo o conteúdo publicado. Esse problema pode custar muito tempo e esforço das pessoas envolvidas nesse gerenciamento"* Adiers (2007).

Ferramentas voltadas à solução deste problema são fundamentais para auxiliar os usuários não-técnicos a adicionar, editar e gerenciar colaborativamente o conteúdo em um site Web.

As primeiras ferramentas visando flexibilizar o desenvolvimento de software foram denominadas de framework. Sommerville (2007) define que "*um framework é uma estrutura genérica que pode ser ampliada para criar subsistemas ou aplicações mais específicas*" (p.283). Em outras palavras, um framework ou arcabouço é uma estrutura de suporte definida em que um outro projeto de software pode ser organizado e desenvolvido. Ele pode incluir programas de suporte, bibliotecas de código, linguagens de script e outros softwares no sentido de colaborar no desenvolvimento de diferentes componentes em um projeto de software.

Côrtes & Lucena (2001) consideram que:

> *Com o surgimento do paradigma da orientação a objetos, a tecnologia adequada para reuso de grandes componentes tornou-se disponível e resultou na definição de frameworks orientados a objetos. Os principais benefícios dos frameworks orientados a objetos decorrem da modularidade, reusabilidade e extensibilidade que eles oferecem aos desenvolvedores.*

Com a evolução das tecnologias, especialmente das tecnologias voltadas à Internet, surgem sistemas que buscam facilitar as técnicas de programação envolvendo frameworks orientados a objetos. Neste contexto surgem os Sistemas Gerenciadores de Conteúdo – SGC

(do inglês Content Management System - CMS), que permitem a criação, armazenamento e administração de conteúdo de forma dinâmica, através de uma interface de usuário via Internet (Gerosa, 2006). O SGC encapsula um ou mais frameworks em uma estrutura que visa anular ou reduzir a tarefa do desenvolvedor de conhecer o framework para desenvolver a partir dele.

Os SGC difundiram-se inicialmente no meio empresarial, tornando-se a ferramenta central da arquitetura de sites de centenas de empresas nacionais, internacionais e multinacionais como, por exemplo, Porsche Brasil e Mitsubishi Polônia. Após a inserção dos SGC no desenvolvimento de sites empresariais, tornaram-se responsáveis, também, pelo desenvolvimento de sites governamentais, diversas prefeituras utilizam SGC em seus sites, por exemplo, Jaraguá do Sul – SC, Tibagi – PR, Alto Alegre – MA, Ilhéus – BA, Manoel Viana – RS, Petrópolis – RJ etc. Além de prefeituras, outros sites ligados a órgãos governamentais utilizam SGC: Secretaria de Ciência e Tecnologia do Piauí (SECTEC), Partido Socialismo e Liberdade (PSol), Tribunal de Contas do Distrito Federal (TCDF) etc.

Nas instituições educacionais, a utilização de SGC é tímida, ela aparece na composição dos sites da universidade de Pernambuco (UPE), Universidade Federal do Tocantins (UFT), University of Ottawa – Canadá e algumas instituições privadas de ensino.

A inserção dos SGC nas universidades se apresenta apenas no tocante a construção do site da universidade ou de áreas específicas, como a página do curso de Engenharia Oceânica da UFRJ, não houve registros de utilização de SGC no desenvolvimento de ambientes virtuais de aprendizagem.

Um software educativo é essencialmente um sistema passível de abordagens de engenharia de software (Sommerville, 2007 e Pressman, 2007). Segundo Pressman (2007), "*um sistema baseado em computador é definido como: Um conjunto de disposições de elementos que é organizado para executar certo método, procedimentos ou controle ao processar informações*". Em outras palavras, "*um sistema é o conjunto intencional de componentes inter-relacionados que funcionam juntos para atingir certo objetivo*" (Sommerville, 2007, p.14).

A produção de um software depende de um conjunto coerente de atividades e etapas bem estabelecidas, que possam ser usadas para explicar diferentes abordagens do desenvolvimento de software. A engenharia de software denomina estes conjuntos de atividades de "*paradigma de engenharia de software*" (Pressman, 2007, p.30) ou "*modelo de processo de software*" (Sommerville, 2007, p.42).

Um paradigma de processo que oriente o desenvolvimento de um software deve ser escolhido baseado na natureza do projeto a ser desenvolvido. Neste contexto destacam-se quatro paradigmas já amplamente estudados, são eles: clássico (Pressman, 2007, Sommerville, 2007), prototipação (Pressman, 2007) ou Evolucionário (Sommerville, 2007), espiral (Pressman, 2007, Sommerville, 2007) e baseado em componentes (Sommerville, 2007).

O paradigma clássico é também conhecido como modelo cascata. Este modelo demanda uma abordagem sistematicamente seqüencial para execução do projeto (Pressman, 2007, p.32). Nas últimas décadas vem sendo criticado sobre a sua aplicabilidade em todas as situações, principalmente na área de desenvolvimento Web. O principal problema dessa abordagem apresenta-se na realidade dos projetos, que raramente seguem um fluxo seqüencial como o modelo sugere. Além, disso é difícil para o gerenciador do projeto definir todos os requisitos explicitamente numa única etapa, especialmente no começo. As vantagens deste modelo consistem na documentação produzida em cada fase e a sua aderência a outros modelos de processo de engenharia.

A prototipação, também conhecida como evolucionária pode ser a melhor abordagem quando durante a elaboração de um projeto, definindo-se um conjunto de objetivos gerais a serem alcançados, não se tem informação detalhada dos requisitos. Em certos casos, o projetista não está certo sobre a eficiência do produto final. Nestes casos, como em muitos outros, o paradigma de prototipação pode ser a melhor abordagem, pois implica na principal vantagem deste processo, que consiste, segundo Sommerville (2007) "*na especificação poder ser desenvolvida de forma incremental, à medida que os usuários compreendem melhor seus problemas*" (p.45). Sommerville (*op. cit.*) aponta as desvantagens deste modelo "n*a possibilidade de corromper a estrutura do software devido às mudanças contínuas. Isto torna a incorporação de mudanças de software cada vez mais difícil e onerosa*" (Sommerville, *op. cit.*).

O modelo espiral foi proposto para unificar as melhores características do modelo clássico com as do modelo prototipal, enquanto que ao mesmo tempo formaliza a inclusão de um novo elemento, a análise dos riscos que nos falta outros paradigmas. Os riscos podem causar problemas no projeto, tal como ultrapassar o cronograma e os custos. Segundo Pressman (2007), "*O modelo espiral usa a prototipação como um mecanismo de redução dos riscos, mas, o que é mais importante, possibilita que o desenvolvedor aplique a abordagem de prototipação a qualquer etapa evolutiva*" (p.38). No entanto "*este modelo não é uma*

panacéia, ele exige considerável experiência na avaliação de riscos, assim se um grande risco não for descoberto, indubitavelmente ocorrerão problemas" (Pressman, *op. cit.*).

O modelo baseado em componentes é apresentado por Sommerville (2007) como um modelo "*baseado na existência de um número significativo de componentes reusáveis. O processo de desenvolvimento do sistema enfoca a integração desses componentes, em vez de desenvolvê-los a partir do zero*" (p.46). Na maioria dos projetos de software existe algum reuso de software, isso ocorre geralmente de forma informal e independentemente do processo de desenvolvimento usado. Nos últimos anos a abordagem denominada engenharia de software baseada em componentes, que conta com reuso, tem emergido e se tornado cada vez mais utilizada. "*A abordagem orientada a reuso depende de uma grande base de componentes reusáveis e algum framework de integração desses componentes*" (Sommerville *op. cit.*). Este desenvolvimento possui a vantagem de reduzir a quantidade de software a ser desenvolvido e consequentemente reduzir os custos e riscos. Como desvantagem apresenta-se o fato do modelo exigir comprometimento com os requisitos dos componentes, o que pode levar a um sistema que não atenda às reais necessidades dos usuários.

Em comum, todos os paradigmas sugerem como etapas no processo de desenvolvimento: análise de requisitos, projeto do sistema, codificação e testes.

Análise de requisitos

A análise de requisito é uma etapa fundamental, pois serão os requisitos definidos nesta etapa os desenvolvidos no sistema, segundo Boehm (1984 *apud* Cysneiros, 2001, p.16) "*quando só detectados depois do software implementado, erros em requisitos de software são até 20 vezes mais caros de se corrigir que qualquer outro tipo de erro*".

Sommerville (2007, p.80), classifica os requisitos em: i) requisitos funcionais e não funcionais, ii) requisitos de usuário, iii) requisito de sistema, iv) especificação de interface. Os requisitos de software são, frequentemente, classificados em requisitos funcionais e não funcionais.

> *Os requisitos funcionais são requisitos que expressam funções ou serviços que um software deve ou pode ser capaz de executar ou fornecer. As funções ou serviços são, em geral, processos que utilizam entradas para produzir saídas.*
> *Os requisitos não funcionais são requisitos que declaram restrições, ou atributos de qualidade para um software e/ou para o processo de desenvolvimento deste sistema. Segurança, precisão, usabilidade, performance e manutenabilidade são exemplos de requisitos não funcionais.*
> *Grosso modo, um requisito funcional expressa algum tipo de transformação que tem lugar no software, enquanto um requisito não funcional expressa*

> *como essa transformação irá se comportar ou que qualidades específicas ela deverá possuir* (Cysneiros, 2001).

Desta forma os requisitos funcionais descrevem as funcionalidades que se espera que o sistema disponibilize, de uma forma completa e consistente. É aquilo que o utilizador espera que o sistema ofereça ou o que o sistema não deve fazer.

Projeto do sistema

O projeto é a primeira etapa da fase de desenvolvimento de qualquer produto ou sistema de software. Ele pode ser definido como "*o processo de se aplicar várias técnicas e princípios ao propósito de se definir um dispositivo, um processo ou um sistema com detalhes suficientes para permitir sua realização física*" (Taylor, 1959 *apud* Pressman, 2007, p.415).

Segundo Pressman (2007, p.416), "*o projeto de software encontra-se no núcleo técnico do processo de engenharia de software, iniciando-se tão logo os requisitos de software tenham sido analisados e especificados*". Para Sommerville (2007),

> *Os requisitos de usuário devem ser escritos em linguagem natural. Pois eles devem ser compreendidos por pessoas que não são especialistas técnicos. Contudo, os requisitos mais detalhados do sistema podem ser expressos de maneira mais técnica. Uma técnica amplamente usada é documentar a especificação do sistema como um conjunto de modelos. Esses modelos são representações gráficas que descrevem, de forma geral, o sistema a ser desenvolvido. Devido às representações gráficas usadas, os modelos geralmente são mais compreensíveis do que descrições detalhadas de sistema em linguagem natural* (p.112).

Os modelos devem confrontar variadas interpretações e declarações, em linguagem natural, e poder suscitar, de forma a proporcionar "*melhor compreensão da entidade real a ser construída*" (Pressman, 2007, p.248).

Neste contexto surgem os conceitos de objetos e modelagem de objetos definidos por Silva (2006): "*objetos são entidades executáveis, instâncias de classes que definem seus atributos e serviços. Modelos de objetos são utilizados quando se pretende adotar o paradigma de orientação a objetos nas atividades de desenho e implementação*".

Desde a década de 1990 vários métodos de representar modelos orientados a objetos foram propostos. Três dos grandes desenvolvedores (Booch, Rumbaugh e Jacobson) integraram suas abordagens sobre este tema e produziram um único método conhecido como linguagem de modelagem unificada (UML – Unified Modeling Language). A UML se tornou padrão para a modelagem orientada a objeto, como a utilizada neste sistema (Booch *et al*, 2000). Segundo Pressman (2007, p.418) A modelagem de um sistema é um processo de

múltiplos passos que se concentra em quatro atributos distintos do programa: i) Projeto Arquitetural, ii) Projeto Procedimental, iii) Projeto de Dados e iv) Projeto de Interfaces.

Codificação

No processo de codificação inserem-se técnicas de desenvolvimento específicas e personalizadas. No contexto do trabalho, inserem-se os sistemas gerenciadores de conteúdo com o objetivo de acelerar o desenvolvimento do software.

Os SGC direcionam o desenvolvimento de um software através da abordagem baseada em componentes ou módulos. Esta abordagem possui algumas vantagens e desvantagens, apresentadas por Gerosa (2006):

> *Para projetar um sistema baseado em componentes é necessário entender os benefícios e dificuldades da tecnologia, além do ferramental disponível. Os benefícios da componentização estão ligados à manutenibilidade, reuso, composição, extensibilidade, integração, escalabilidade, entre outros. As dificuldades são separadas em dificuldades do desenvolvimento para componentização (construção dos componentes e da infra-estrutura) e dificuldades do desenvolvimento com componentização. As dificuldades do desenvolvimento para componentização estão ligadas ao esforço inicial de análise, projeto e desenvolvimento, enquanto as dificuldades do desenvolvimento com componentização estão ligadas ao esforço despendido no entendimento dos componentes e das ferramentas envolvidas, à perda de flexibilidade, à dependência de terceiros e à adaptação do processo de desenvolvimento.*

A componentização ou modularidade no software de computador tem sido adotado há quase quatro décadas. A arquitetura de software incorpora a modularidade dividindo-se em componentes separadamente nomeados e endereçáveis, denominados módulos, que são integrados para atender aos requisitos ditados pelo problema (Pressman, 2007, p.427). Pressman (*op. cit.*) afirma que "*Modularidade é o atributo de software que permite que um programa seja intelectualmente administrável*".

Pressman (2007) define que "*um projeto modular reduz a complexidade, facilita a mudança (um aspecto crítico da capacidade de manutenção do software) e resulta numa implementação mais fácil ao estimular o desenvolvimento paralelo de diversas partes de um sistema*" (p.438).

Teste

> *Durante as fases de definição e desenvolvimento anteriores, o engenheiro de software tenta construir o software, partindo de um conceito abstrato para uma implementação tangível. Agora, surge a fase de testes. A atividade de*

> *testes é um processo que poderia ser visto como destrutivo, em vez de construtivo* (Pressman, 2007, p.787).

A etapa de testes torna-se fundamental para que sejam desconstruídos possíveis erros de codificação e requisitos implementados baseados em interpretações equivocadas da real necessidade do cliente. Isto constitui o processo natural de evolução do conhecimento do software, como apresentado por Sommerville (2007, p.107), "*a evolução de requisitos durante o processo de engenharia de requisitos e após a entrada de um sistema em operação, é inevitável. À medida que a definição dos requisitos se desenvolve, uma compreensão maior das necessidades é obtida*".

Capítulo III

DESENVOLVIMENTO

Neste capítulo é apresentado o desenvolvimento do ambiente EVA observando-se dois momentos distintos:

- O inicial, quando se fez uso de um paradigma de desenvolvimento visando à pré-testagem do sistema didático EVA;
- Posteriormente, na implementação de uma versão beta do sistema didático EVA.

3.1 – Natureza do Problema

Neste ponto é possível Retomar as questões expostas como de interesse quando se assumiu o desenvolvimento do EVA. Estas questões serão descritas a seguir com a intenção de identificar cada um dos pontos que consistiam em bugs pontuais e sistêmicos e que requeriam modificações imediatas.

Natureza da Interface

- Ergonomia das telas de apresentação – A apresentação dos conteúdos se dá em telas visualmente incômodas, “causando estresse ao olho humano” (Sommerville, 2007, p.248).
- Navegabilidade entre as telas de apresentação – A navegabilidade entre as páginas não favorecia a aprendizagem, já que os menus não eram apresentados de forma organizada.

Natureza Estrutural

- Impossibilidade de separar os alunos e professores em grupos – adicionava-se um prefixo aos nomes dos usuários para facilitar a distinção visual entre os membros do grupo, porém isso impossibilitava que um aluno participa-se de mais de um grupo.
- Erro no tratamento de datas – as datas armazenadas em banco de dados após execução de um evento que necessitasse desta ação, eram despadronizadas e imprecisas, ou seja, não representava a informação desejada.
- Impossibilidade de cadastrar visitantes – a condição de visitante não era possível de ser implementada, visto que o AVEC não previa a atribuição de permissões

específicas a grupos de atores distintos. Para um usuário conhecer o AVEC era necessário receber uma senha com privilégios de professor, aluno ou administrador.

- Falta de ferramentas wysiwyg – os campos de texto destinados à digitação de textos que exigem formatação mais detalhada, como a resposta ou a correção de passos de um Estudo de Caso, necessitam de um editor wysiwyg para que haja a possibilidade desta formatação, no AVEC esta ferramenta é inexistente.
- Falta de ferramentas para repositórios de arquivos – Não havia uma ferramenta de repositório de arquivos, para um professor disponibilizar um material era necessário solicitar ao administrador que criasse um link estático para o arquivo.
- Flexibilidade nula para adição de novos recursos ou melhorias – A estrutura do sistema AVEC não era modular, assim a intervenção para a inclusão de um recurso ou a melhoria de algum existente torna-se oneroso e arriscado.
- Inexistência de distinção entre os atores – O sistema não realizava a distinção entre os diferentes atores do sistema, assim um aluno poderia acessar os mesmos recursos que um professor.
- Despadronização – No AVEC não havia padronização dos códigos, por exemplo, algumas telas eram montadas em CSS, outras em HTML e outras com a própria linguagem de programação.

Natureza da Documentação

- Padrões de projeto – não há registros de documentos especificando os padrões de projeto adotados e modelagem do sistema AVEC.
- Estrutura de banco de dados – Não há diagramas que demonstrem a estrutura de banco de dados do AVEC.

3.2 – Sistema Inicial

O desenvolvimento inicial do sistema EVA se deu na adoção de um paradigma da engenharia de software para orientar este processo. Esta definição baseou-se no contexto do desenvolvimento, que apontava uma quantidade de requisitos bem definidos para desenvolvimento em um período total de 7 meses antes de sua primeira utilização. A maioria dos projetos de software possui um levantamento de requisitos confuso e impreciso, porém para o ambiente EVA isto não se verificou de forma acentuada em virtude da experiência com o ambiente anterior, do qual se manteve boa parte dos requisitos.

Em virtude das falhas encontradas no ambiente AVEC e da opção feita pela criação do ambiente EVA, estabeleceu-se o desenvolvimento de alguns requisitos melhorados em um período de tempo objetivando seu teste inicial. Desta forma não se pretendeu esboçar os requisitos de todo o sistema EVA de forma idealizada, visto que seria uma tarefa prematura. Pretendeu-se reformular os requisitos do AVEC, desenvolvê-los de forma confiável, segura e testá-los. Após o teste, remodelar o escopo instrucional do ambiente a partir de informações obtidas com os usuários do pré-teste. Deste modo evita-se modelagem equivocada de requisitos.

De forma resumida, o cenário inicial apresentava-se com uma parcela bem clara de requisitos a ser codificada de forma seqüencial, em um período de tempo pré-estabelecido, visando um pré-teste ao término do ciclo de codificação. Para este cenário verificou-se a maior compatibilidade com o modelo clássico ou em cascata que, segundo Pressman (2007, p.32-34) e Sommerville (2007, p.44-45), é composto de 5 etapas básicas: i) Análise, ii) Projeto, iii) Codificação, iv) Teste e v) Manutenção (Figura 3.1).
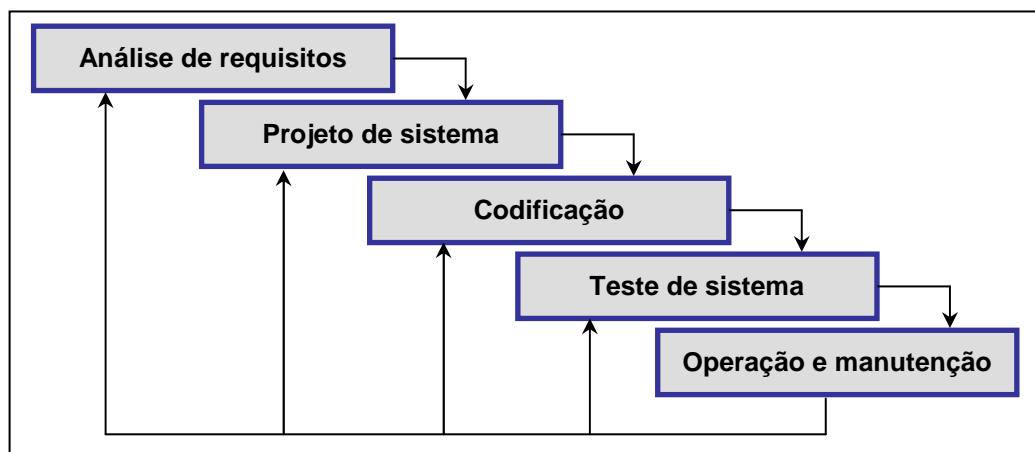
Figura 3.1 – Ciclo de vida clássico ou em cascata

Para o desenvolvimento do ambiente EVA denominou-se a etapa “teste de sistema” como Pré-testagem e a etapa “Operação e manutenção” como Remodelagem de escopo.

A execução das etapas de “Análise de requisitos”, “Projeto de sistema” e “codificação” se deu em um período de 7 meses, em seguida foi realizado um pré-teste do ambiente durante 2 meses. Ao término do pré-teste foi feito, no período de 1 mês, a remodelagem do escopo instrucional do sistema. O quadro 3.1 apresenta a divisão do tempo realizada para estas etapas em um calendário dividido em 20 quinzenas.

ETAPA	Quinzena																			
	1°	2°	3°	4°	5°	6°	7°	8°	9°	10°	11°	12°	13°	14°	15°	16°	17°	18°	19°	20°
Análise de requisitos																				
Projeto do sistema																				
Codificação																				
Pré-teste																				
Remodelagem																				

Quadro 3.1 – Calendário de execução das etapas do projeto

As seções seguintes abordam de forma detalhada cada uma das etapas.

3.2.1 - Análise de Requisitos

A análise dos requisitos é "*o processo de comunicação entre os clientes e usuários do software com os desenvolvedores de software*" (Sommerville, 2007, p77). Na análise de requisitos do sistema EVA os clientes foram os professores orientadores, que também são usuários e os desenvolvedores, a equipe de tecnólogos do CEFET-Campos. A partir de diversas reuniões entre os professores e a equipe de desenvolvimento, foram levantados os requisitos funcionais e não-funcionais necessários para implementação e funcionamento do EVA.

Para atender as necessidades da aplicação do modelo de ABC e a utilização da Internet, foram estabelecidos os requisitos funcionais listados abaixo, embora o principal requisito funcional do sistema EVA seja: o sistema deve fornecer telas, recursos suficientes e apropriados para que os Estudos de Caso sejam concretizados.

Além deste, outros requisitos funcionais, mais específicos, são expostos no quadro 3.2 para os diferentes atores do sistema.

Ator	Requisito Funcional
Professor	• Deve ser capaz de abrir tema no fórum e gerenciar as mensagens postadas nele. • Deve ser capaz de publicar avisos. • Deve ser capaz de associar um Estudo de Caso da biblioteca de Estudo de Casos a um de seus grupos. • Deve ser capaz de abrir e fechar uma sala de chat. • Deve ser capaz de corrigir as soluções aos passos enviadas pelos alunos, atribuindo-lhes uma resposta e a aprovação ou requisição de mudanças. • Deve ser capaz de gerenciar repositório de arquivos. • Deve ser capaz de gerenciar notas no mural • Deve ser capaz de manter resenhas.
Aluno	• Deve ser capaz de acessar, buscar e fazer download dos materiais didáticos contidos no repositório de arquivos de um de seus grupos. • Deve ser capaz de responder aos passos do Estudo de Caso. • Deve ser capaz de pesquisar resenhas para a auxiliar na resposta ao passo 2.
Administrador	• Deve ser capaz de gerenciar usuários, p. ex, cadastrar, editar, excluir e associar aos grupos. • Deve ser capaz de publicar avisos. • Deve ser capaz de armazenar e gerenciar um Estudo de Caso na biblioteca de Estudo de Casos. • Deve ser capaz de abrir tema no fórum e gerenciar as mensagens postadas nele. • Deve ser capaz de gerenciar repositório de arquivos. • Deve ser capaz de gerenciar configurações do sistema. • Deve ser capaz de gerenciar grupos de estudo. • Deve ser capaz de gerenciar Estudos de Caso, p. ex, cadastrar, editar, excluir. • Deve ser capaz de gerenciar notas no mural • Deve ser capaz de abrir e fechar uma sala de chat. • Deve ser capaz de manter resenhas. • Deve ser capaz de gerenciar os módulos e seus blocos. • Deve ser capaz de gerenciar as permissões aos diferentes níveis de acesso. • Deve ser capaz
Comum a todos	• Deve ser capaz de se cadastrar, preencher e editar seus dados cadastrais. • Deve ser capaz de efetuar seu login e ser reconhecido como professor, aluno ou administrador. • Deve ser capaz de comunicar-se com outros usuários através de mensagens particulares mediadas pelo sistema. • Deve ser capaz de enviar mensagem em um dos temas do fórum relacionado a um grupo. • Deve ser capaz de enviar mensagem em uma das salas de chat relacionada a um grupo. • Deve ser capaz de acessar portfólio. • Deve ser capaz de consultar notas no mural de recados. • Deve ser capaz de consultar o catálogo de usuários cadastrados no sistema.

Quadro 3.2 – Requisitos funcionais para os diferentes atores do ambiente

Estes requisitos funcionais exprimem o comportamento compreendido que o EVA deveria apresentar ao fim de seu desenvolvimento. Todavia é necessário especificar os requisitos não funcionais, pois não estão apenas relacionados com o sistema a ser desenvolvido, são afetos também às necessidades do usuário, às restrições de orçamentos e prazos, às políticas organizacionais, aos regulamentos de segurança etc. A figura 3.2 representa os tipos de requisitos não funcionais.
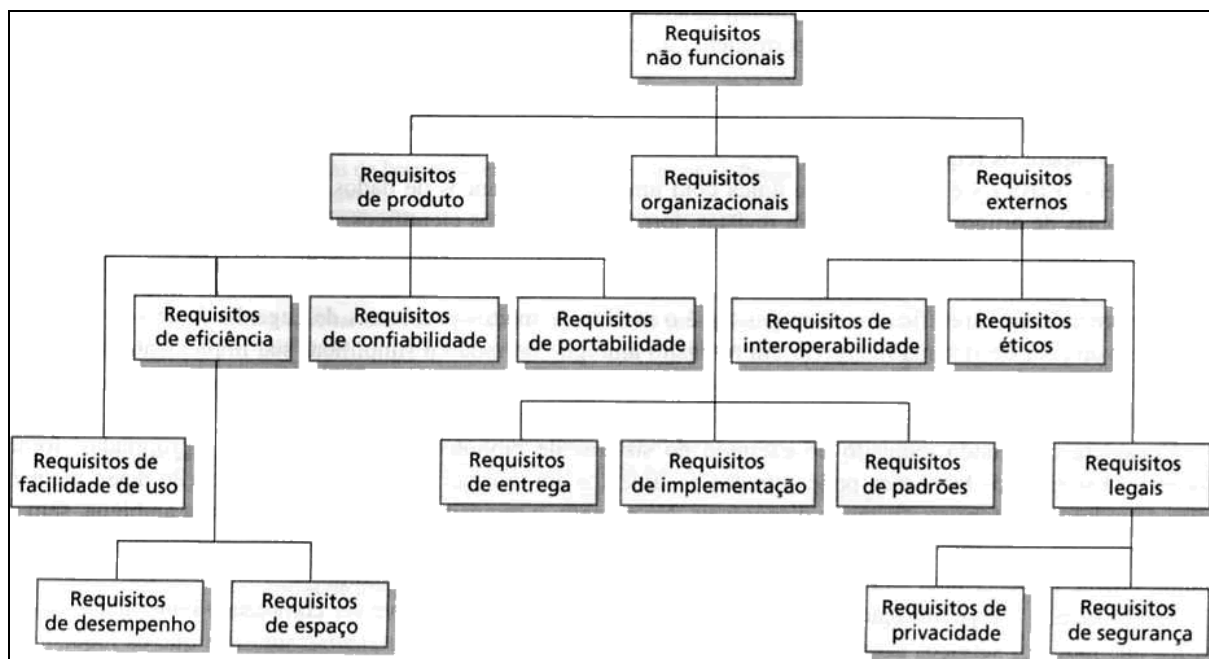
Figura 3.2 – Tipos de requisitos não funcionais (Sommerville, 2007, p.82)

Alguns tipos de requisitos não funcionais foram considerados fundamentais na análise de requisitos:

Facilidade de uso: O sistema deve possuir relativa facilidade de uso para que o sistema não se contraponha ao processo de aprendizagem do aluno, impondo barreiras tecnológicas.

Confiabilidade: O sistema deve transmitir confiança aos usuários que o utiliza. Confiança relativa à preservação dos dados cadastrais ou dos Estudos de Caso, por exemplo, uma solução de um passo, que deve ser registrada e mantida inalterada para leitura do professor.

Portabilidade: O sistema deve permitir ser portável a qualquer sistema operacional sem falhas.

Entrega: Este é um dos requisitos mais importantes, pois estabelece os prazos para a entrega do sistema. Baseado no cronograma inicial apresentado anteriormente, o sistema deveria ser entregue com seus requisitos funcionais mínimos necessários ao funcionamento do sistema EVA implementados decorridos 5 meses do fim da fase de projeto.

Implementação: O sistema deveria ser implementado utilizando tecnologias opensource e de conhecimento dos desenvolvedores ou de rápida aprendizagem, além de permitir sua hospedagem nos servidores das instituições em que o projeto estava se desenvolvendo. Neste sentido a linguagem de programação adotada foi o PHP[3] na sua

[3] PHP (Hypertext Preprocessor") é uma linguagem de script Open Source de uso geral, muito utilizada e especialmente guarnecida para o desenvolvimento de aplicações dinâmicas Web embutidas no HTML (www.php.net).

abordagem orientada a objetos e o banco de dados MySQL[4] ou PostgreSQL[5]. O PHP é uma linguagem de relativa facilidade de aprendizagem, é livre, permite desenvolvimento orientado a objetos e as instituições possuem servidores dedicados a execução de aplicações escritas nesta linguagem. O banco de dados optou-se pelo MySQL ou PostgreSQL por serem banco de dados estáveis e livres.

3.2.2 – Projeto do Sistema

O projeto do sistema EVA consistiu em modelar os requisitos funcionais e não-funcionais visando proporcionar maior compreensão dos mesmos na etapa de codificação. Os projetos criados para o sistema EVA seguem a divisão proposta por Pressman (2007) e Sommerville (2007):

Projeto Arquitetural: definir os grandes componentes estruturais do software e seus relacionamentos.

Projeto Procedimental: refinar e detalhar a descrição procedimental dos componentes estruturais da arquitetura.

Projeto de Dados: projetar a estrutura de armazenamento de dados necessária para implementar o ambiente.

Projeto de Interfaces: projetar a interface de apresentação do sistema com os usuários.

3.2.2.a – Projeto Arquitetural

Até esta etapa do projeto a questão do hardware ainda não foi tratada, já que na fase de análise pressupôs-se tecnologia perfeita. Este é o momento para resolver como este modelo ideal irá rodar em uma plataforma restrita.

Xavier & Portilho (1995) e Sommerville (2007, p.163), orienta-nos a determinar, durante o projeto arquitetural, as seguintes estruturas:

- Os componentes de hardware para as máquinas clientes;
- Os componentes de hardware para as máquinas servidoras;
- A localização ou localizações dos processos;
- A localização ou localizações dos dados físicos;
- A distribuição geográfica dos requisitos;

---

[4] O MySQL tornou-se o software open source de banco de dados mais popular do mundo, devido sua consistência, rapidez, performance, alta confiabilidade e fácil uso. Ele é utilizado em todos os continentes por desenvolvedores individuais e por grandes organizações (www.mysql,org).

[5] PostgreSQL é um ponderoso sistema open source de banco de dados relacional. Ele possue mais de 15 anos de desenvolvimento ativo e uma arquitetura que apresenta forte confiabilidade, integração e exatidão (www.postgresql.org).

- As estratégias de sincronização para dados distribuídos geograficamente e
- A configuração e o número de camadas cliente-servidor.

O requisito não funcional de portabilidade define o EVA como um sistema multiplataforma, portanto as máquinas clientes devem acessar o EVA através de um computador com sistema operacional Windows, Linux, Solaris, Mac OS etc. Os componentes de hardware utilizados nas máquinas clientes para executar o sistema operacional devem ser suficientes para acessar o EVA.

Em virtude das definições contidas no requisito não funcional de implementação, o sistema operacional de um servidor para executar o EVA deveria ser o Linux, por ser um sistema opensource e mais eficiente para execução da linguagem PHP com o Apache. Quanto à configuração de hardware, o sistema EVA deveria ser capaz de funcionar satisfatoriamente com as mesmas configurações de hardware e rede recomendadas para o funcionamento das principais distribuições linux para servidor (red hat, slackware, debian etc).

Todos os processos e dados físicos seriam armazenados em um único servidor. Está opção anula a possibilidade de projetar a distribuição geográfica dos requisitos e as estratégias de sincronização dos dados. Porém não impede que se adote uma postura de distribuição de dados em vários servidores e viabilize uma estrutura de sincronização de dados posteriormente.

A arquitetura cliente-servidor segue duas abordagens: arquitetura cliente servidor de hardware e de software.

A arquitetura cliente-servidor de hardware adotada para o EVA deveria manter a integridade dos dados do negócio em uma máquina hospedeira central acessível pelas máquinas clientes. Nesta arquitetura o acesso ao EVA não deveria necessitar de vínculos decorrentes de software intermediários, instalados nos computadores dos usuários. Para atender está definição, optou-se pela arquitetura de hardware cliente-servidor em duas camadas, a camada servidor e a camada clientes. Esta arquitetura de hardware contempla as especificações estabelecidas, pois apresenta múltiplos clientes fazendo requisições a um único servidor (figura 3.3). Deve-se ressaltar que a opção por esta arquitetura de hardware não inviabiliza a inserção de servidores intermediários, alterando, assim, a arquitetura para em 3 camadas ou em *n* camadas.
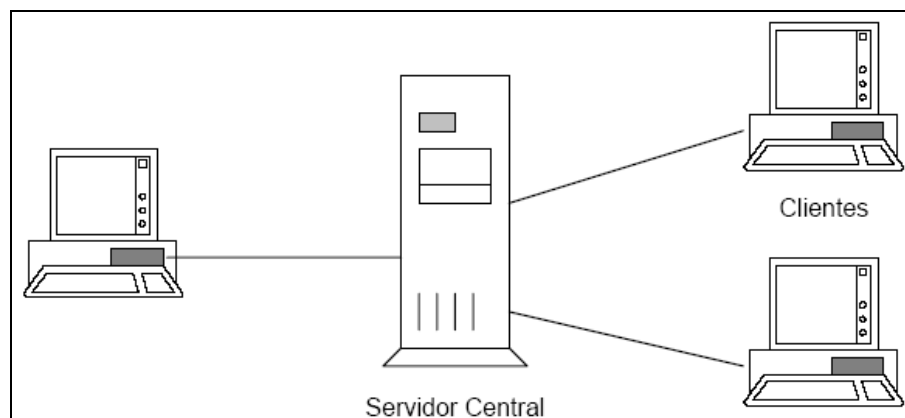
Figura 3.3 – Arquitetura de hardware cliente-servidor em duas camadas

Para a arquitetura cliente-servidor de software optou-se por dividir os componentes de software da aplicação em três camadas: apresentação, negócio e dados.

Camada de Apresentação: É a camada mais externa do sistema de software ou simplesmente interface. Esta camada interage diretamente com o usuário. Possui a função de capturar estímulos de eventos externos e realizar alguma edição dos dados de entrada. Encarrega-se também de apresentar respostas da camada de lógica do negócio ao usuarios.

Camada de Lógica do Negócio: Esta camada contém as funções e regras de todo o negócio. Nesta camada não deve existir interface para o usuário e seus dados devem ser voláteis, ou seja, para que algum dado seja armazenado deve-se utilizar a camada de dados.

Camada de Gerência de Dados: Esta camada recebe as requisições da camada de negócios e seus métodos executam as requisições em um banco de dados, que gerencia requisições concorrentes de acesso às bases de dados, processa e devolve à camada de negócios os dados requisitados.

As relações lógicas entre as três camadas são apresentadas na figura 3.4.
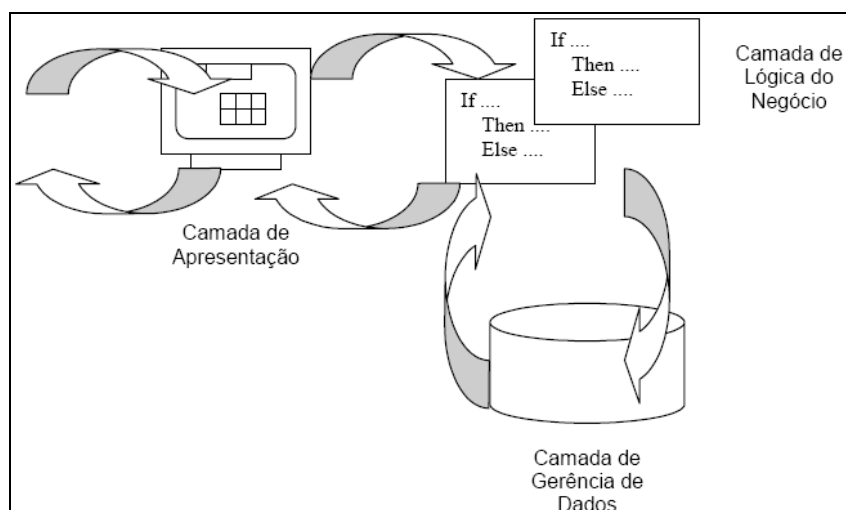
Figura 3.4 – Arquitetura de software cliente-servidor em três camadas

Devido o paradigma híbrido adotado, o modelo cascata baseado em componentes, a camada de lógica do negócio ou camada lógica foi dividida em componentes, denominados módulos, de forma que os módulos se tornassem independentes e possibilitassem sua substituição sem afetar a arquitetura do sistema. A estrutura modular idealizada para o EVA é apresentada na figura 3.5 através de um diagrama de pacotes da UML.
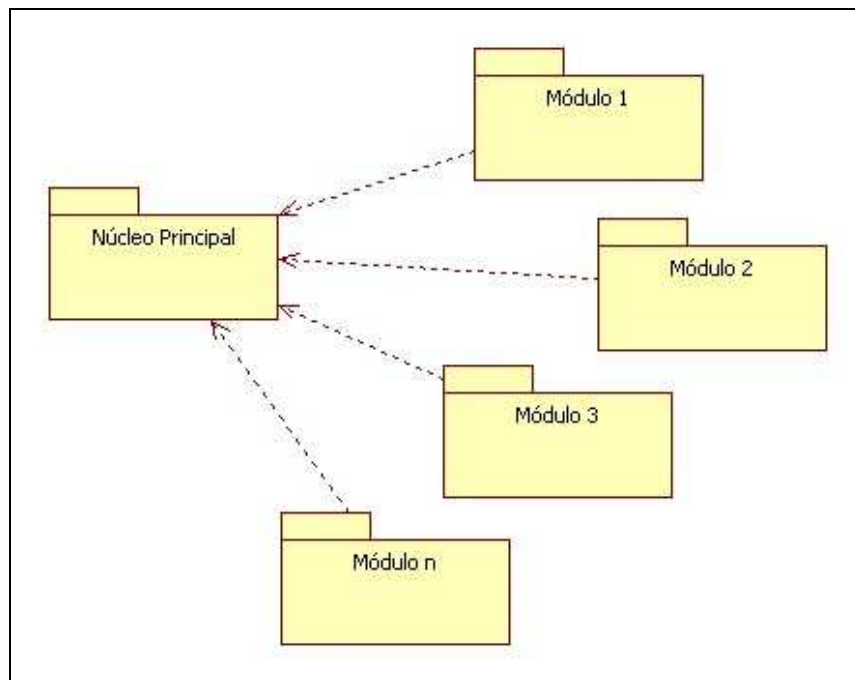
Figura 3.5 – Diagrama de pacotes que representa a arquitetura do sistema EVA

Esta arquitetura possui um módulo principal e a partir dele outros módulos são acoplados. "*O módulo principal é o conjunto de características do sistema que fornecem apoio para o propósito essencial do sistema*" (Sommerville, 2007, p.516).

3.2.2.b – Projeto Procedimental

O projeto procedimental do sistema EVA visou elucidar a questão da modularidade, para isso foi necessário não projetar os requisitos separadamente, mas sim agrupá-los em módulo, componentes maiores contendo funcionalidades correlatas. A especificação dos módulos para o sistema EVA depara-se com uma questão fundamental no projeto conceitual de modularidade: "*como decompor uma solução de software para obter o melhor conjunto de módulos?*" (Pressman, 2007, p.437). Uma sugestão é proposta por Parnas (1972 apud Pressman, 2007): "*os módulos devem ser especificados e projetados de tal forma que as informações (procedimento e dados) contidas num módulo sejam inacessíveis a outros módulos que não tenham necessidade de tais informações*" (p.437).

Assumindo esta sugestão, buscou-se agrupar em cada módulo, requisitos que se comunicam de forma lógica entre si, mas que não necessitam de requisitos de outros módulos.

Para se definir quais módulos seriam criados para a arquitetura do sistema além do módulo principal, utilizou-se alguns dos conceitos presentes no ambiente AVEC. Apesar deste não possuir uma arquitetura em módulos, foi possível compreender e interpretar suas funções para especificar quais módulos o sistema EVA deveria possuir, são eles:

- Avisos – Espaço destinado a lembretes e mensagens que organizam o estudo.
- Fórum – Ferramenta de comunicação onde os professores e alunos podem discutir acerca de um tema aberto, exclusivamente, pelo professor.
- Chat – Ferramenta de comunicação onde os professores e alunos podem discutir acerca de um tema aberto, exclusivamente, pelo professor.
- Repositório de arquivos – Espaço destinado a armazenar materiais didáticos. Cada estudo de caso possui uma biblioteca de materiais.
- Mural – Ferramenta dedicada aos usuários que queiram exibir algum recado para o grupo.
- Portfólio – Espaço utilizado para disponibilizar as propostas de solução indicadas como as mais factíveis para cada Estudo de Caso utilizado.
- Catálogo de usuário – Espaço destinado à gerência dos usuários cadastrados no sistema.
- Estudos de Caso – Espaço onde professores e alunos trabalham os Estudos de Caso e se relaciona, de forma lógica, com os demais módulos associados ao estudo.

Uma forma de modelar o projeto dos procedimentos de um módulo é através de diagramas UML de casos de usos e de atividades, sendo este, necessário para oferecer uma especificação precisa do processamento, inclusive a seqüência de eventos, pontos de decisão exatos e operações repetitivas, ou seja, dos fluxos de atividades dos casos de uso (Pressman, 2007, p.435-437).

A seguir apresentada-se sequencialmente a descrição dos casos de uso de cada módulo acompanhado do diagrama de casos de uso do módulo. Cada caso de uso possui um diagrama de atividades, que representa o fluxo dos dados durante sua execução. Os diagramas de atividades estão disponibilizados no anexo eletrônico 1.

Principal: A figura 3.6 apresenta os casos de uso do módulo central (principal). Os casos de uso deste módulo implementam recursos próprios, por exemplo, “logar”, “gerenciar usuários” e proporciona as ligações lógicas com os demais módulos, “gerenciar módulos”.
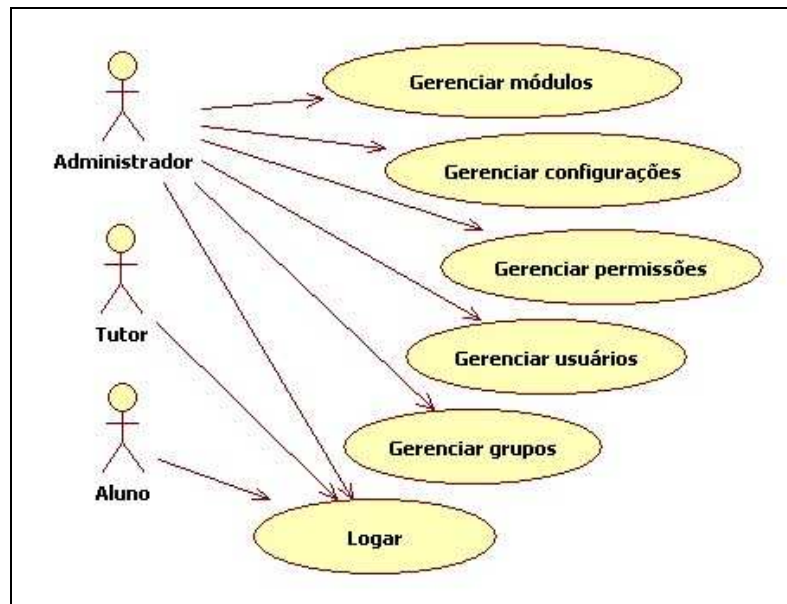
Figura 3.6 – Casos de uso do módulo principal

Mural: A figura 3.7 apresenta os dois casos de uso do módulo mural. Este módulo funciona como mural de avisos e possui os administradores e professores como gerenciadores dos avisos e todos os atores podem consultar os avisos.
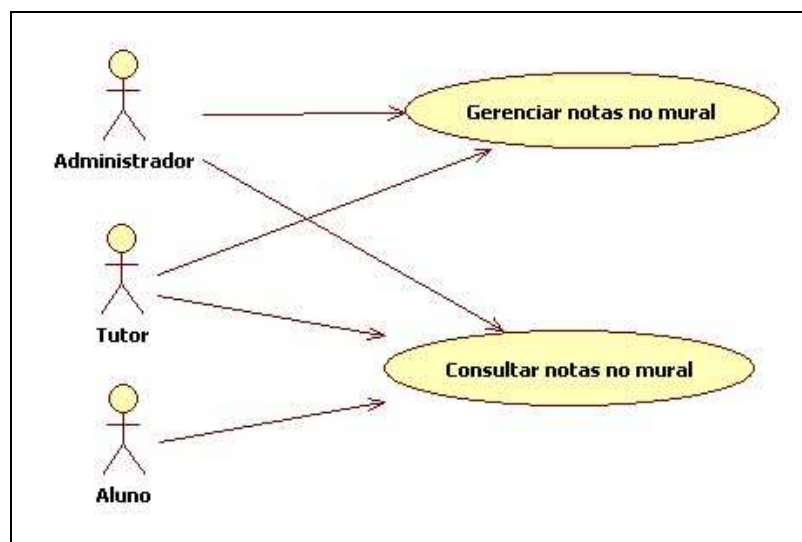
Figura 3.7 – Casos de uso do módulo mural

Repositório de arquivos: A figura 3.8 Apresenta os casos de uso que os atores podem realizar no módulo repositório de arquivos. Os casos de uso “manter cadastro de arquivo” destinados aos administradores e professores e “obter arquivos” destinado a todos os atores.
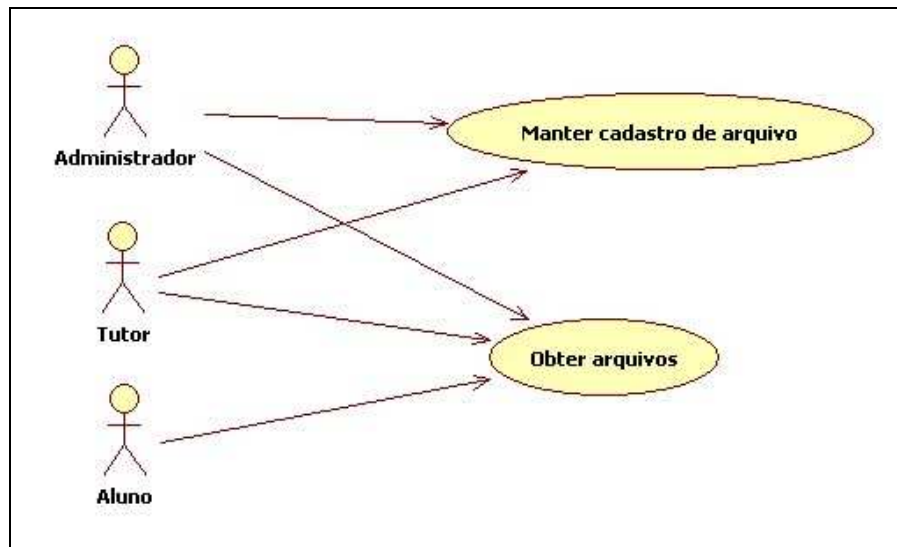
Figura 3.8 – Casos de uso do módulo repositório de arquivos

Catálogo de usuário: A figura 3.9 apresenta o caso de uso único deste módulo, “consultar catálogo”, que permite consulta aos usuários cadastrados no ambiente.
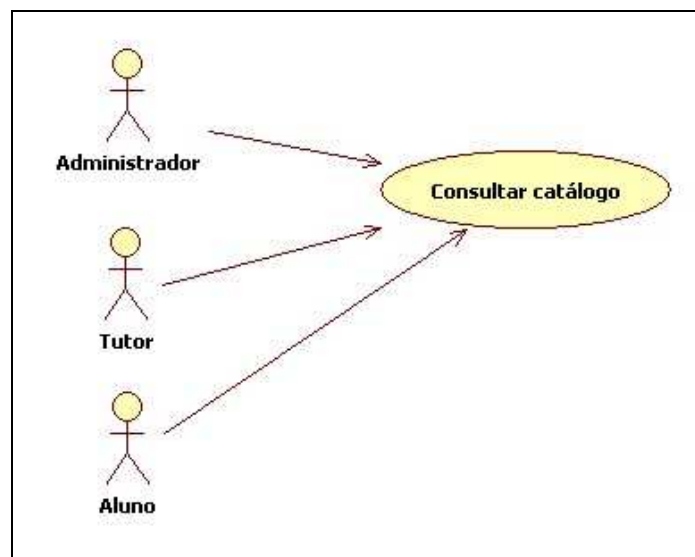
Figura 3.9 – Caso de uso do módulo catálogo de usuários

Chat: A figura 3.10 apresenta os casos de uso do módulo de chat, nela Apresenta-se os casos de uso dos administradores e professores que são “criar sala” e “fechar sala”, sendo que esse implica em o professor ou administrador ter selecionar sala. Todos os atores podem selecionar uma sala de chat, podendo estender ao envio de uma mensagem, “enviar mensagem”.
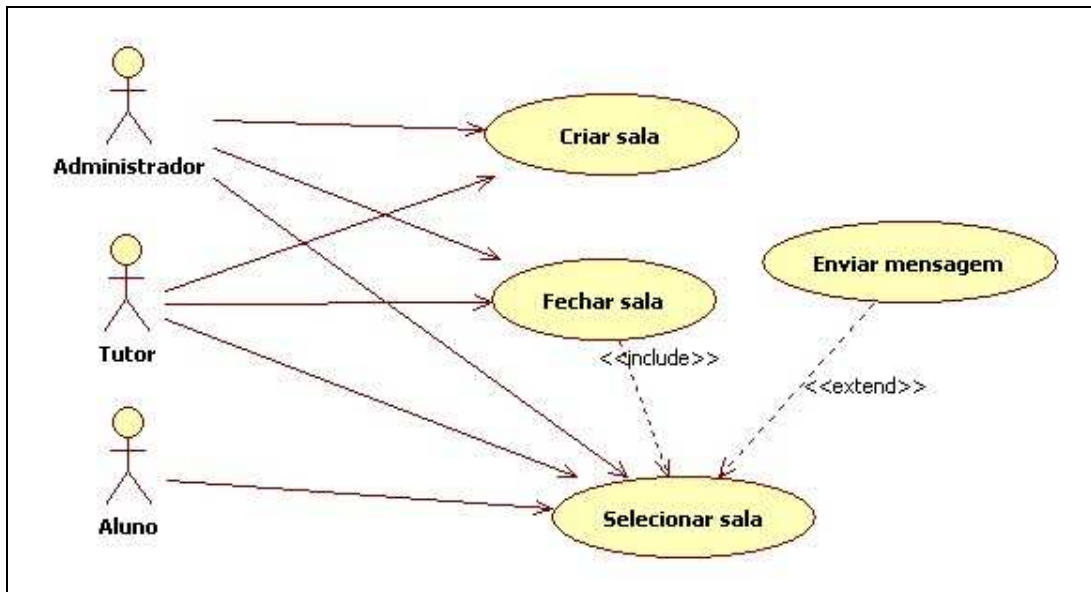
Figura 3.10 – Casos de uso do módulo de chat

Biblioteca de resenhas: A figura 3.11 apresenta os dois casos de uso que os atores podem realizar no módulo repositório de arquivos. Os casos de uso são: “manter cadastro de resenhas e pesquisar resenhas”, sendo o primeiro acessível ao professor e ao administrador e o segundo a todos os atores.
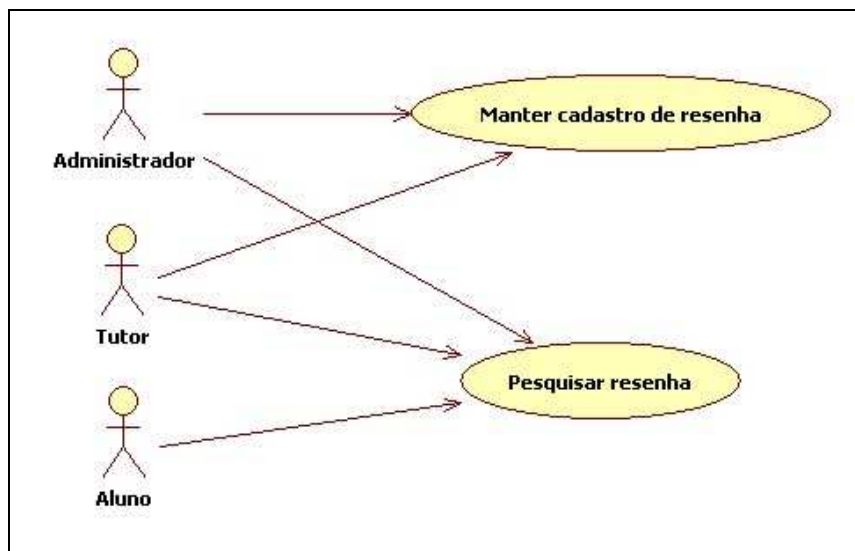
Figura 3.11 – Casos de uso do módulo biblioteca de resenhas

Fórum: A figura 3.12 apresenta os casos de uso do módulo fórum divididos em duas partes. A primeira é referente aos casos de uso na perspectiva do professor e do administrador e a segunda na do aluno. Os atores “selecionar um assunto”, “selecionar um tema” e “enviar mensagem”, porém os alunos não podem “gerenciar tema” e “gerenciar assunto” (restrito ao administrador) e não podem “incluir tema” (restrito ao professor e administrador).
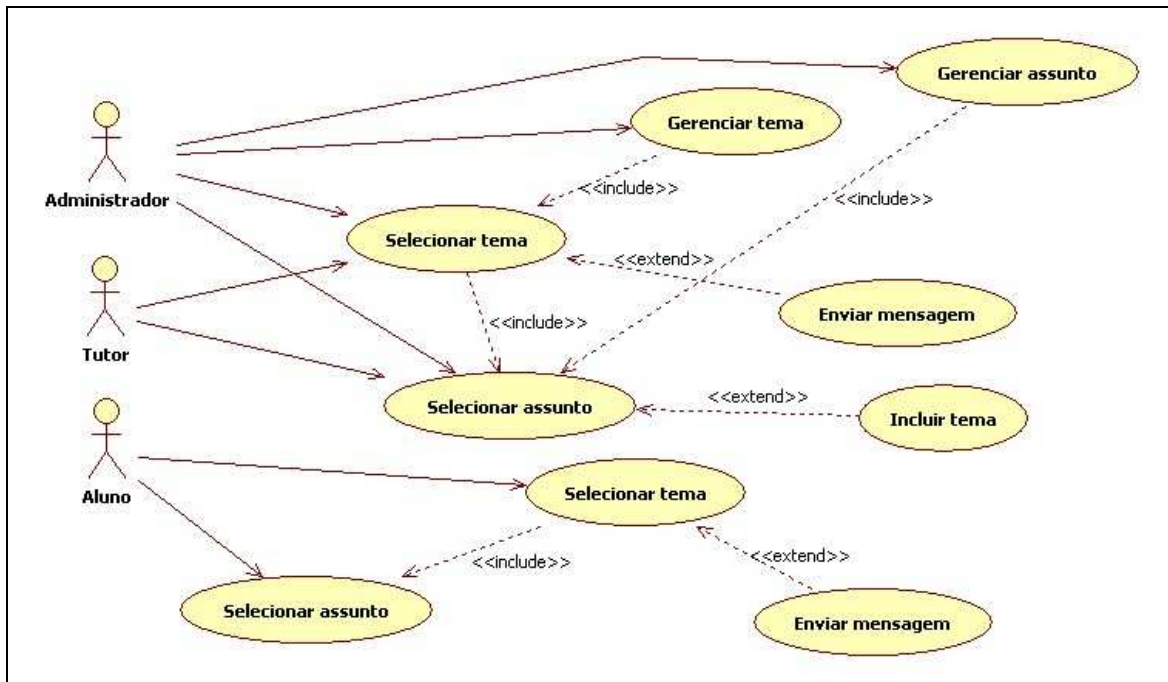
Figura 3.12 – Casos de uso do módulo de fórum

Avisos: A figura 3.13 apresenta os dois casos de uso que os usuários podem realizar no módulo repositório de avisos. Os casos de uso são: “manter cadastro de avisos” e “visualizar avisos”, sendo o primeiro acessível ao professor e administrador e o segundo a todos os atores.
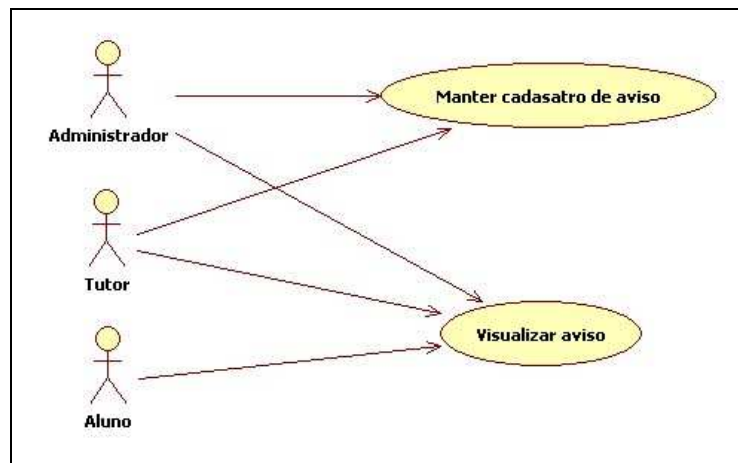
Figura 3.13 – Casos de uso do módulo de avisos

Estudos de Caso: A figura 3.14 apresenta os casos de uso do módulo de Estudos de Caso. Observa-se que todos os atores podem “selecionar estudo de caso”, que inclui “selecionar grupo de estudo”. Além destes, os professores podem “associar estudo de caso ao grupo” e “corrigir passos”. Os administradores podem “manter estudos de caso”. Os alunos podem responder passos.
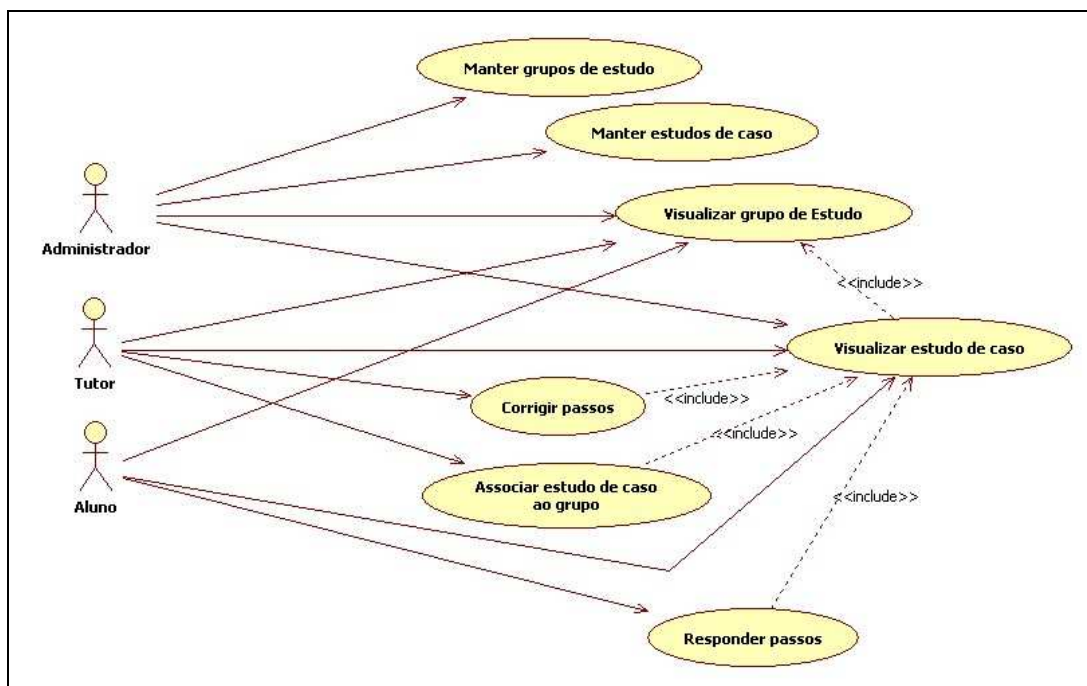
Figura 3.14 – Casos de uso do módulo de Estudos de Caso

Portfólio: A figura 3.15 apresenta os dois casos de uso do módulo de portfólio. São eles: “manter portfólio” e “consultar portfólio” de usuário, ambos acessíveis a todos os atores do ambiente.
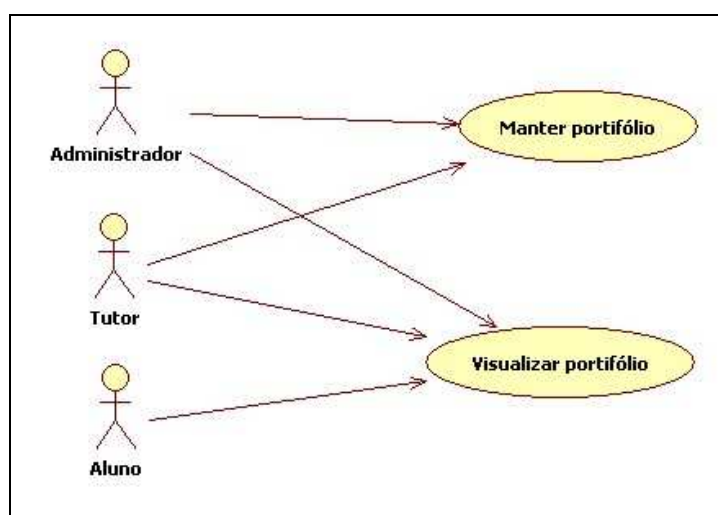
Figura 3.15 – Casos de uso do módulo de portfólio

3.2.2.c – Projeto de Dados

Um aspecto fundamental da fase do projeto consiste em estabelecer de que forma serão armazenados os dados do sistema (Xavier & Portilho, 1995). Neste sentido, representou-se o modelo conceitual dos dados do sistema EVA através de um diagrama de classes da UML (figura 3.16).

Figura 3.16 – Diagrama de classe do sistema

As classes são representadas pelos retângulos de forma que a linha superior contenha o nome da classe, a linha intermediária os atributos (características do objeto gerado pela classe) e na linha inferior os métodos (operações que a classe desempenha para o sistema).

O diagrama de classe apresentado na figura 3.16 define todas as classes que o sistema EVA necessita possuir para seu funcionamento, porém não representam como os objetos serão persistidos, ou seja, salvos em banco de dados.

Os bancos de dados orientados a objeto não têm uma forte aceitação dos desenvolvedores de software, já que perdem o fundamento formal matemático, ao contrário do modelo relacional, e às vezes conduz a fraqueza na sustentação da consulta. Isto fez com que até os dias atuais prevaleça o modelo de banco de dados relacional, por exemplo, a empresa norte-americana oracle é a terceira maior empresa de desenvolvimento de software do mundo, apesar de possuir um único produto de software, o banco de dados relacionais oracle®.

Um dos maiores problemas enfrentados pelos atuais desenvolvedores envolve adequar os projeto de dados, que são orientados a objetos, para os bancos de dados relacionais predominantes. Segundo o requisito não-funcional de implementação, os bancos de dados para persistência dos dados do sistema EVA seriam o MySQL ou PostgreSQL, ambos relacionais.

A adaptação do projeto orientado a objetos para o relacional incluiu mapear as classes a fim de determinar as tabelas e os campos. Durante a conversão considerou-se que nem sempre uma classe representa uma tabela e vice-versa e que um atributo de uma classe pode ser mapeado para ‘0’ ou mais colunas de uma tabela, já que em uma classe nem todos os atributos são persistidos. Neste sentido, foi necessário representar a estrutura dos dados através de diagramas voltados à representação de estruturas relacionais.

Um diagrama utilizado para tal representação é o diagrama de entidade relacionamento – DER, utilizado para representar a estrutura lógica de banco de dados de forma abstrata. Uma extensão dos DER são os diagramas de estrutura de dados – DED, que proporcionam a representação das tabelas de dados e seus relacionamentos em modelos conceituais de dados.

O DED do sistema EVA é apresentado na figura 3.17.
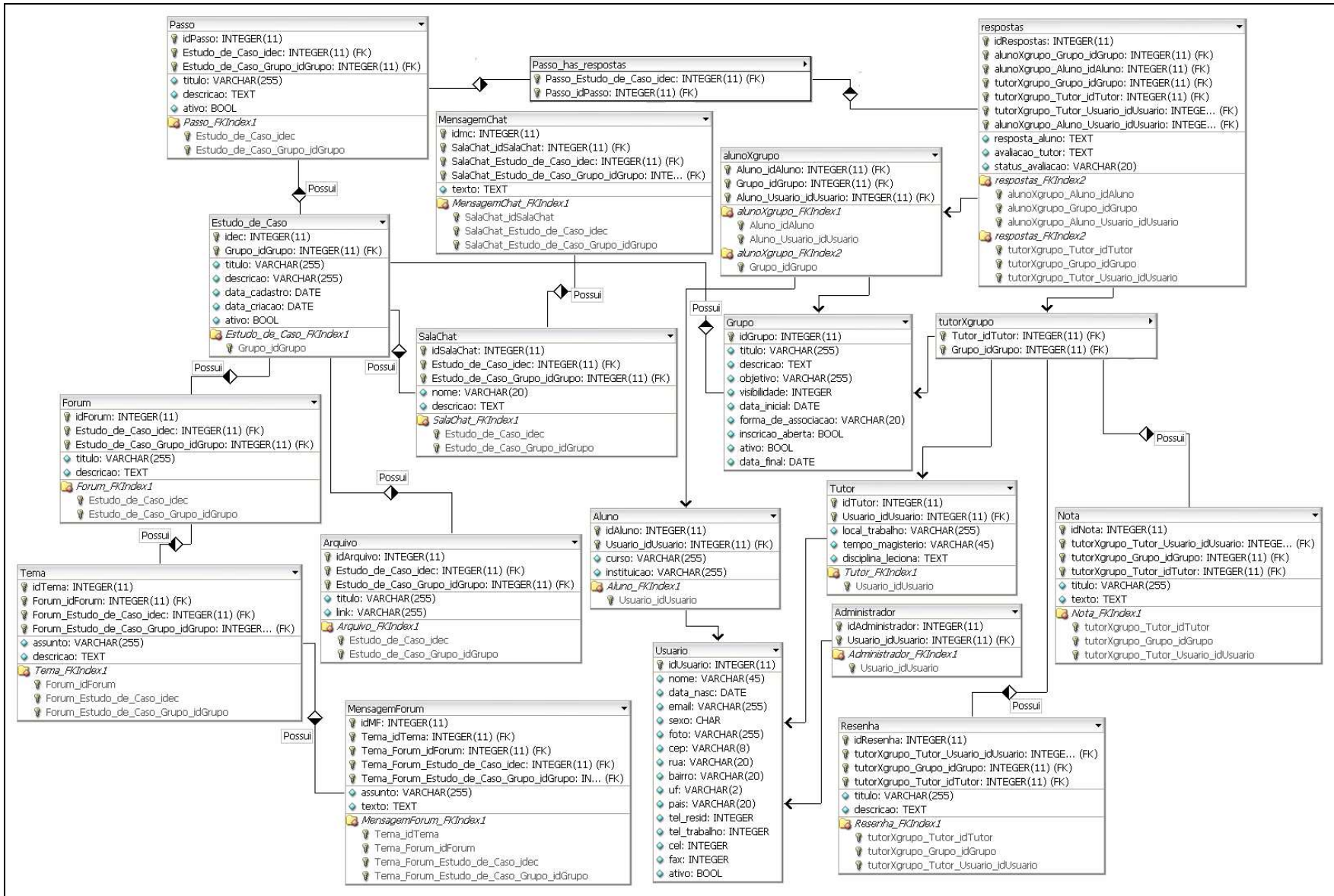
Figura 3.17 – Diagrama estrutura de dados do sistema

No gráfico os retângulos representam as tabelas, cada linha do retângulo uma coluna na tabela de dados e as linhas entre as tabelas representam os relacionamentos.

3.2.2.d – Projeto de Interfaces

O projeto da interface para o EVA foi parte essencial de todo o projeto do sistema. No projeto de interface buscou-se uma interface com o usuário projetada para combinar as habilidades, experiências e expectativas dos usuários previstos. O princípio da familiaridade do usuário nos sugere que os usuários do sistema didático EVA não devem ser forçados a se adaptar a uma interface porque a implementação desta é conveniente. Desta forma algumas sugestões foram adotadas durante o projeto da interface.

As sugestões seguidas neste projeto foram propostas por Alves & Pires (2002) no trabalho "*A usabilidade em software educativo: princípios e técnicas*", em que entendem usabilidade como a simplicidade e facilidade com que a interface pode ser utilizada. A partir do trabalho de Alves & Pires (*op. cit.*) é possível destacar alguns pontos que deveriam ser cumpridos ou evitados na interface do sistema EVA.

Os principais pontos que o sistema deveria cumprir são:

- Usar etiquetas ALT para todas as imagens, especialmente as de navegação.
- Usar texto negro em fundo branco sempre que possível para uma legibilidade maior.
- Usar cores de fundo planas ou com texturas, mas que sejam extremamente sutis.
- Assegurar que o texto se encontra numa cor que se possa imprimir (não branco).
- Colocar o menu de navegação numa localização consistente em cada página do site.
- Usar localizações familiares para as barras de navegação.
- Usar um design adequado para que nunca seja necessário recorre ao scroll horizontal.
- Usar um eixo de simetria para centrar o texto numa página.
- Encorajar o uso do scroll através da colocação de uma imagem que fique semi-escondida no fundo da página.

Deveria-se tentar evitar que:

- As etiquetas ALT sejam reduzidas (especialmente para uma pequena imagem de tamanho fixo).
- Mostrar texto estático em azul ou a sublinhado.
- Usar o negrito ou maiúsculo para texto longo.
- Deixar espaços em branco muito longos – a leitura torna-se mais difícil.

- Obrigar o utilizador a fazer scroll para encontrar informação importante, especialmente botões ou links de navegação.
- Usar travessões para separar horizontalmente diferentes tipos de conteúdos.
- Alternar frequentemente entre texto centrado e alinhado à esquerda. A maior parte do texto deve estar alinhada à esquerda.
- Fixar a resolução das páginas a mais de (800X600). Páginas de grande resolução podem obrigar o utilizador a fazer scroll horizontal.

3.2.3 – Codificação

Todos os passos de engenharia de software descritos até este ponto estão dirigidos a um objetivo final: traduzir representações de software para uma forma que pode ser "entendida" pelo computador. Esta etapa se configura como a etapa de codificação propriamente dito do sistema. Segundo Pressman (2007, p.676), a codificação é a conseqüência natural do projeto.

3.2.3.a – Prioridade de Codificação

Após análise e modelagem dos requisitos, o requisito não funcional de entrega estabelecia que grande parcela dos requisitos funcionais deveria ser codificada em 5 meses, prazo definido para pré-teste do ambiente.

Para Pressman (2007, p.102), "*As estimativas de custo e de esforço de software jamais serão ciências exatas. Muitas variáveis – humanas, técnicas, ambientais e políticas – podem afetar o custo final do software e o esforço aplicado para desenvolvê-lo*".

As estimativas de custo e esforço citadas por Pressman nos revertem em estimativas de tempo e esforço. Neste sentido foi necessário dimensionar coerentemente as funcionalidades mínimas a serem implementadas no período de 5 meses, de forma que fossem suficientes para a execução do modelo ABC e não extrapolassem o referido período.

A fase de projeto revelou as funcionalidades do ambiente como sendo: i) mural, ii) repositório de arquivos, iii) catálogo de usuários, iv) chat, v) biblioteca de resenhas, vi) fórum, vii) avisos, viii) Estudos de Caso, ix) portfólio e x) principal. Entretanto para implementação no referido período priorizou-se as seguintes funcionalidades:

Principal: por ser o módulo central de controle, deveria ser o primeiro módulo a ser desenvolvido.

Estudos de Caso: trata-se do módulo pedagógico central do ambiente e, portanto, fundamental a aplicação do modelo de ABC.

Fórum: o fórum é a principal ferramenta de comunicação do ambiente, assim sua implementação foi necessária para suporte ao módulo de Estudos de Caso.

Biblioteca de resenhas: a biblioteca é o local onde armazenaria os materiais relativos as resenha, necessária no passo 2, assim a biblioteca de resenhas constituiu-se como necessária.

Avisos: necessário para deixar algum aviso relativo a prazos, notas etc aos alunos.

Chat: necessário como ferramenta de comunicação alternativa ao fórum.

3.2.3.b – Ferramenta de Apoio ao Desenvolvimento

No desenvolvimento do sistema EVA optou-se pela utilização de um sistema gerenciador de conteúdo para acelerar o processo de codificação. Esta opção foi feita com o objetivo de explorar a capacidade destas ferramentas de possibilitar o reuso de uma grande estrutura de componentes para a construção de um novo sistema em tempo reduzido, possibilitando a entrega dentro dos prazos estabelecidos.

Esta abordagem baseada na utilização de sistemas gerenciadores de conteúdo recai sabre o paradigma da engenharia de software baseado em componentes já descrito. A adoção deste modelo para o desenvolvimento do EVA, não implica em descartar o paradigma inicial definido, o modelo em cascata. Pressman (2007, p.44) alerta para a necessidade de não abordar os paradigmas como soluções alternativas, e sim como soluções complementares: "*Em muitos casos os paradigmas podem e devem ser combinados de forma que as potencialidades de cada um possam ser obtidas em um único projeto*" (Pressman, *op. cit.*).

Definiu-se, portanto, que a codificação das funcionalidades mínimas do EVA incorporaria o modelo baseado em componentes (figura 3.18) ao modelo em cascata.
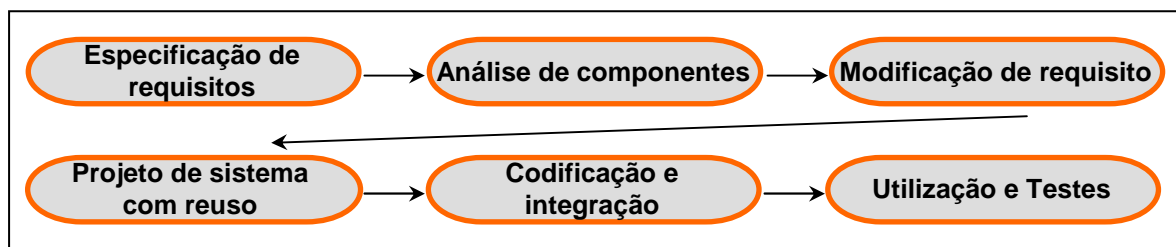
Figura 3.18 – Paradigma de engenharia de software baseado em componentes

Para combinar os modelos em cascata (figura 3.1) e baseado em componentes (figura 3.18) foi necessário depreender que os dois possuem seu início na análise e especificação de requisitos e os testes do modelo em cascata equivalem à Utilização e Testes do baseado em componentes, assim a seqüência intermediária do modelo baseado em componentes: análise de componentes → modificação de requisitos → projeto de sistema com reuso →

desenvolvimento e integração, pode substituir sem prejuízo a etapa de codificação do modelo em cascata, resultando no modelo combinado, híbrido apresentado na figura 3.19, ao qual denominou-se de modelo em cascata baseado em componentes.
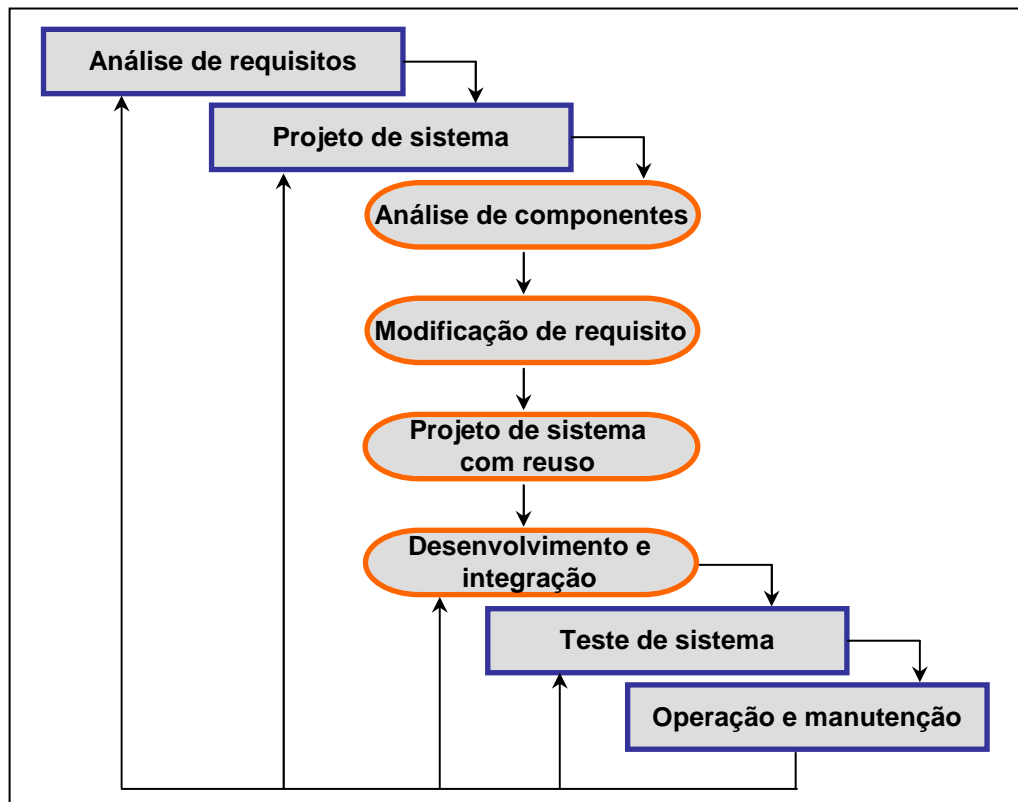
Figura 3.19 – Paradigma híbrido: modelo em cascata baseado em componentes

Este modelo híbrido orienta as duas primeiras etapas, de análise e projeto, de acordo com o modelo em cascata. Ao concluir a etapa de projeto, ocorre a alteração do paradigma, o modelo híbrido inicia as etapas referentes ao modelo baseado em componentes: análise de componentes, modificação de requisito, projeto com reuso e integração. Após a etapa de integração, o modelo em cascata é retomado para a execução das etapas de teste de sistema e manutenção.

3.2.3.c – Sistemas Gerenciadores de Conteúdo

Nos últimos anos vem surgindo o conceito de SGC, ferramenta que busca facilitar as técnicas de programação envolvendo frameworks. Segundo Adiers (2007), os SGC surgem para suprir as necessidades das organizações de fácil e rápido gerenciamento de portais. Muitas organizações usam em seus site Web algum SGC para comunicação com seus colaboradores, publicidade, publicação de notícias, eventos, etc. No entanto os SGC evoluíram de uma realidade empresarial para uma definição mais ampla, como a proposta por Gerosa (2006): "*O SGC é um tipo de sistema utilizado para construir portais na web, que*

oferecem suporte à interação com os usuários". Aí se insere a proposta de utilização de um SGC no desenvolvimento de sistemas educativos, visto que se baseiam na interação entre os usuários.

Com a proposta de acelerar o desenvolvimento foi estabelecido que o sistema seria codificado a partir de um sistema gerenciador de conteúdo. Para a escolha do SGC foi utilizado como referência o site cmsmatrix[6], destinado a realizar análises comparativas entre SGC. O cmsmatrix possui, atualmente, 815 SGC cadastrados, dentre eles Gerosa (2006) e Adiers (2007) definem alguns, que vêm se tornando populares por oferecer flexibilidade para o administrador do portal compor as páginas, configurá-las e posicioná-las. São eles: **Mambo**[7], **XOOPS**[8], **PHP-Nuke**[9], **Joomla**[10], **Plone**[11]**, Lumis**[12], **DotNetNuke**[13] e **Xaraya**[14].

É necessário situar os CMS com a data de sua escolha, primeiro semestre de 2005. Nesta data os desenvolvedores iniciais do mambo retiraram-no da condição de opensource para torná-lo pago. A última versão do Mambo ainda opensource foi adaptada por uma nova equipe de desenvolvedores, constituindo-se em um projeto paralelo de desenvolvimento. O novo CMS surgido desta mudança de política do Mambo foi denominado de Joomla. Desta forma, não foi considerado o Joomla na escolha, pois se tratava de um CMS recém-criado e sem estabilidade, mesmo sendo hoje considerado por diversos autores um dos melhores CMS livres existentes.

O requisito não funcional de implementação estabelece que a linguagem de programação utilizada para o desenvolvimento do sistema EVA seria o PHP e banco de dados MySQL ou PostgreSQL e que qualquer sistema utilizado em apoio ao desenvolvimento deveria ser opensource. Assim o Mambo, Lumis, Plone e DotNetNuke foram descartados. O mambo e o Lumis por serem pagos. O Plone por possuir como linguagem de programação Python. O DotNetNuke por possui como linguagens de programação ASP.NET 2.0 e VB.NET.

Os gerenciadores de conteúdo listados acima e que passaram pela clivagem do requisito não funcional de implementação foram: PHP-Nuke, Xaraya e XOOPS. A escolha de

---

[6] http://www.cmsmatrix.org
[7] http://www.manbofoundation.org
[8] http://www.xoops.org
[9] http://www.phpnuke.org
[10] http://www.joomla.org
[11] http://www.plone.org
[12] http://www.lumis.com.br
[13] http://www.dotnetnuke.com
[14] http://www.xaraya.com

um depende especificamente da finalidade do uso, já que cada SGC possui suas vantagens e desvantagens, assim tornou-se necessária uma análise comparativa entre eles.

A comparação entre os SGC foi realizada no site CMSMatrix, de onde extraímos tabelas abordando os seguintes assuntos: características gerais, segurança, suporte, facilidade de uso, desempenho, gerenciamento, interoperabilidade, flexibilidade e aplicações. A tabela completa encontra-se no anexo 1. A seguir são apresentados sequencialmente tabelas com os itens considerados fundamentais na escolha do SGC para o desenvolvimento do ambiente EVA. Para cada item, nas tabelas, é colocado "Sim" se o SGC atender àquele item, "Não" se não atender, "Limitado" se atender parcialmente e "plug-in" se atender mediante instalação de um plug-in.

A tabela 3.1 contém as características gerais dos 3 SGC selecionados, são elas:

- Servidor Web – Servidores de aplicação Web compatíveis com o SGC.
- Banco de dados – Banco de dados compatíveis com o SGC.
- Sistema operacional – Quais sistemas operacionais são compatíveis com o SGC.
- Acesso de root – Possibilita acesso administrativo?
- Acesso à kernel – É necessário estar logado como administrador para acessar o kernel do SGC?

Características gerais	**PHP-Nuke**	**Xaraya**	**XOOPS**
Servidor Web	Apache, IIS	Qualquer servidor habilitado para PHP	Apache, IIS
Banco de dados	MySQL, Postgres, mSQL, Interbase, Sybase	MySQL, MySQLi, PostgreSQL, SQLite	MySQL 4.23.xx or later
Sistema operacional	Qualquer	Qualquer	Qualquer
Acesso de Root	Sim	Não	Sim
Acesso à kernel	Sim	Não	Sim

Tabela 3.1 – Itens gerais sobre os SGC avaliados

A tabela 3.2 contém os itens relativos à segurança do SGC, são elas:

- Captcha: Possui o recurso de captcha[15]?
- Permissões granulares – O SGC permite atribuir e reconhecer permissões específicas para cada grupo de atores?
- Verificação de e-mail – O SGC envia mensagem para o email do usuário contendo uma senha ativação com o objetivo de verificar se o email é válido?

[15] CAPTCHA é o acrônimo da expressão em inglês "Completely Automated Public Turing test to tell Computers and Humans Apart" (teste de Turing público completamente automatizado para diferenciar entre computadores e humanos).

- Histórico de login – O SGC permite registrar usuários online e quando?
- Gerenciamento de sessão – O SGC permite que o administrador tenha acesso e gerencie as variáveis de sessão?
- Notificação de problema – O SGC fornece algum mecanismo (e-mail, mensagens instantâneas, celular, etc) para alertar os administradores quando se detecta um problema?

Segurança	**PHP-Nuke**	**Xaraya**	**XOOPS**
Captcha	Não	Plug-in	Plug-in
Permissões Granulares	Limitado	Sim	Sim
Verificação de E-mail	Não	Sim	Sim
Histórico de login	Não	Não	Plug-in
Gerenciamento de Sessão	Não	Sim	Limitado
Notificação de Problema	Não	Não	Limitado

Tabela 3.2 – Itens sobre segurança dos SGC avaliados

A tabela 3.3 contém os itens relativos ao suporte oferecido pelos desenvolvedores e utilizadores do SGC, são elas:

- Manuais – Existem livros ou outros documentos disponíveis para o SGC?
- Suporte – Pode-se obter suporte de organizações comerciais ou não para o SGC?
- Treinamento – Existem organizações comerciais que ofereçam treinamento para o SGC?
- Comunidades de desenvolvimento – Existem comunidades online grátis de desenvolvedor especializados no SGC?

Suporte	**PHP-Nuke**	**Xaraya**	**XOOPS**
Manuais	Não	Limitado	Sim
Suporte	Não	Sim	Sim
Treinamento	Não	Sim	Não
Comunidades de Desenvolvimento	Sim	Sim	Sim

Tabela 3.3 – Itens sobre suporte dos SGC avaliados

A tabela 3.4 contém os itens relativos à facilidade de uso proporcionada pelo SGC, são elas:

- URLs amigáveis – O SGC possui URLs amigáveis (não há grande quantidade de símbolos e números e são normalmente curtas)?
- Redimensionamento de imagens – O SGC possui a capacidade de redimensionar imagens carregadas pelos usuários, evitando-se que o ele precise utilizar um editor externo?
- Template – O SGC possui sistema de templates para controlar o layout das páginas?

- Editor WYSIWYG – O SGC possui um editor WYSIWYG[16]?

Facilidade de uso	PHP-Nuke	Xaraya	XOOPS
URLs amigáveis	Não	Sim	Plug-in
Redimensionamento de imagens	Não	Não	Plug-in
Template	Não	Sim	Sim
Editor WYSIWYG	Não	Plug-in	Plug-in

Tabela 3.4 – Itens sobre facilidade de uso dos SGC avaliados

A tabela 3.5 contém os itens relativos ao desempenho do SGC, são elas:

- Cache de página – O SGC permite o armazenamento de conteúdos das páginas em cache para que seja exibido quando solicitado novamente?
- Balanceamento de carga – O SGC permite a realização de balanceamento de cargas entre múltiplos servidores?
- Replicação de banco de dados – O SGC pode replicar banco de dados para melhor escalabilidade?

Desempenho	PHP-Nuke	Xaraya	XOOPS
Cache de Página	Não	Sim	Sim
Balanceamento de Carga	Não	Sim	Sim
Replicação de Banco de Dados	Não	Não	Não

Tabela 3.5 – Itens sobre desempenho dos SGC avaliados

A tabela 3.6 contém os itens relativos ao gerenciamento permitido pelo SGC, são elas:

- Administração online – O SGC pode ser totalmente gerenciado através de um navegador Web?
- Ferramenta workflow – Existe um sistema de workflow integrado ao SGC?
- Temas / Skins – O SGC proporciona mecanismo de modificação de estilos e tamplates da interface?
- Lixeira – Existe um sistema de lixeira no SGC, que permita aos administradores recuperar o conteúdo que tenha sido removido do sistema?
- Estatísticas Web – O SGC possui algum sistema de estatísticas para exibir informações como telas/itens de conteúdo visualizados pelo usuário, o número de usuários por período de tempo etc.?

[16] WYSIWYG é o acrônimo da expressão em inglês "What You See Is What You Get" (o que você vê é o que você tem). O editor WYSIWYG permite que o usuário formate um texto sem requerer conhecimento de linguagens de estilo e marcação como HTML, CSS, XML ou XSL.

Gerenciamento	**PHP-Nuke**	**Xaraya**	**XOOPS**
Administração online	Sim	Sim	Sim
Ferramenta workflow	Não	Plug-in	Não
Temas / Skins	Sim	Sim	Sim
Lixeira	Não	Plug-in	Não
Estatísticas Web	Sim	Plug-in	Plug-in

Tabela 3.6 – Itens sobre gerenciamento dos SGC avaliados

A tabela 3.7 contém os itens relativos à interoperabilidade oferecida pelo SGC, são elas:

- RSS – O SGC possibilita sistema RSS/XML feeds para exportar conteúdo e ser republicada em outros sites?
- Suporte FTP – O SGC permite aos usuários carregar conteúdo interno e/ou arquivos via FTP?
- XHTML – O SGC segue as especificações W3C para XHTML?

Interoperabilidade	**PHP-Nuke**	**Xaraya**	**XOOPS**
RSS	Não	Sim	Sim
Suporte FTP	Não	Não	Sim
XHTML	Não	Sim	Não

Tabela 3.7 – Itens sobre interoperabilidade dos SGC avaliados

A tabela 3.8 contém os itens relativos à flexibilidade do SGC, são elas:

- Suporte CGI – O SGC pode executar em modo CGI para fins de desenvolvimento?
- Reuso de conteúdo – O SGC permite reuso de conteúdo (não copiado, mas reutilizados de um local para outro do sistema)?

Flexibilidade	**PHP-Nuke**	**Xaraya**	**XOOPS**
Suporte CGI	Sim	Sim	Não
Reuso de conteúdo	Não	Sim	Plug-in

Tabela 3.8 – Itens sobre flexibilidade dos SGC avaliados

A tabela 3.9 contém os itens relativos às aplicações já desenvolvidas para os SGC analisados, são elas:

- Fórum – O SGC possui ferramenta de fórum?
- Chat – O SGC possui ferramenta de chat?
- Gerenciamento de contatos – O SGC possui mecanismos de gerenciamento de contatos?
- Gerenciamento de documentos – O SGC permite gerir arquivos offline?
- Calendário de eventos – O SGC possui ferramenta para exibição de eventos e calendário?

- Ferramenta de busca – O SGC possui mecanismo de busca integrado que pode indexar o conteúdo gerido e permitir ao usuário buscar o conteúdo indexado?
- Gerenciamento de FAQ – O SGC possui ferramenta para organizar as questões mais freqüentes?

Aplicações	**PHP-Nuke**	**Xaraya**	**XOOPS**
Fórum	Sim	Plug-in	Sim
Chat	Não	Plug-in	Plug-in
Gerenciamento de Contatos	Não	Plug-in	Sim
Gerenciamento de Documentos	Não	Plug-in	Plug-in
Calendários de Eventos	Não	Plug-in	Plug-in
Ferramenta de Busca	Não	Plug-in	Sim
Gerenciamento de FAQ	Sim	Plug-in	Sim

Tabela 3.9 – Aplicações já desenvolvidas para os SGC avaliados

Com base nas comparações estabelecidas entre os SGC optou-se pelo SGC XOOPS. Dois pontos foram fundamentais na escolha: O XOOPS e o PHP-Nuke possuem acesso administrativo, mas o Xaraya não. O XOOPS e o Xaraya possuem atribuições de permissões distintas a grupos distintos, mas no PHP-Nuke este recurso é limitado. Nestes dois requisitos fundamentais, o XOOPS se apresentou a melhor escolha.

Outras análises reforçaram a opção pelo SGC XOOPS: os atributos referentes a suporte, o XOOPS possui várias comunidades de desenvolvedores em português, manuais e suporte, diferentemente dos demais, especialmente o PHP-Nuke.

Nos itens relativos à facilidade de uso o XOOPS, também se destaca por permitir a inserção de plug-in para solucionar problemas de redimensionamento de imagens e WYSIWYG, por exemplo. O XOOPS, assim como o Xaraya, mas diferente do PHP-Nuke, possui cache de páginas, que agiliza o processo de carregamento de uma página.

Quanto as aplicações já desenvolvidas para o SGC, o XOOPS e Xaraya atendem grande parte das aplicações pretendidas para o EVA, com fórum, chat etc. Verifica-se que o PHP-Nuke possui apenas aplicações de FAQ e fórum.

Desta forma, o SGC selecionado para o desenvolvimento do sistema EVA foi o XOOPS.

3.2.4 – O XOOPS

O XOOPS é uma ferramenta de gestão de conteúdos para criação e administração de portais dinâmicos para a Web. Criado pelo japonês Kazumi Ono e pelo chinês Goghs Cheng, conta hoje com centenas de colaborados em todo o mundo. É escrito sobre a linguagem PHP e utilizando o banco de dados MySQL, possui seu código fonte aberto e está disponível sob a licença GNU General Public License (GPL), ou seja, é livre para uso, modificações e distribuição (Gerosa, 2006).

O XOOPS é um dos mais populares SGC da atualidade, devido aos recursos que oferece e pelo suporte. Em português o suporte se dá através das comunidades de desenvolvedores. Existem dois livros publicados sobre o XOOPS, um em inglês, *Building Websites with XOOPS: a step-by-step tutorial* (Atwal, 2006) e outro em italiano, *XOOPS: Creare siti dinamici con facilità* (Silvestrini, 2007).

Um site em XOOPS é customizado pelo administrador, que adiciona ou remove módulos através do ambiente administrativo. Cada módulo do XOOPS oferece um conjunto de blocos pré-definidos para exibição das saídas de dados, que são utilizados para compor a interface com o usuário. Alguns módulos já disponíveis na instalação do XOOPS são Notícias, Fórum, Enquete, Links, Downloads, Headlines, FAQ, Parceiros (banners), Usuários e Fale Conosco. Porém centenas de outras são disponibilizadas nas comunidades de desenvolvedores em todo o mundo.

Esta estrutura modular destaca a compatibilidade do SGC com a arquitetura projetada para o sistema EVA.

Módulos:

Os módulos são partes fundamentais de qualquer site XOOPS. Na prática, todos os sites em XOOPS são instalados com um essencial e sofisticado módulo, o módulo [system][17]. Apesar de poucas telas de interação com o usuário, ele possui um núcleo interno com poderosas ferramentas de controle e gerenciamento. É através deste módulo que um administrador pode configurar o site.

Os módulos formam a estrutura dos sites e apresentam o conteúdo, muitas das vezes ele apresenta o conteúdo em blocos de apresentação. Módulos são normalmente construídos para trabalhar em conjunto com o módulo [system] do XOOPS.

[17] Neste trabalho adotaremos que os nomes técnicos dos módulos serão apresentados entre colchetes.

Todo módulo desenvolvido para o XOOPS pode herdar os estilos e templates do layout de apresentação do site, portanto torna-se completamente integrado com as demais páginas do site. Ela também pode herdar outras funções e recursos globais, tais como o acesso de grupos selecionados, permissões diversas, comentários etc. Estas funções são acopladas de modo que o site XOOPS funcione como uma unidade totalmente inter-relacionada que é fácil de gerir pontualmente ou globalmente.

O quadro 3.3 descreve a estrutura geral de um módulo, apresentando os diretórios e arquivos presentes na maior parte dos módulos XOOPS.

Estrutura	Descrição
admin index.php menu.php administracion.php	**admin:** Arquivos da administração do módulo ficam neste diretório. O acesso a esta pasta deve ser restringido. **-admin_header.php:** Verificação de permissões. Inclusões do **mainfile.php** e dados do idioma são feitas aqui. **-index.php:** Módulo de administração de conteúdo deve estar aqui. **-menu.php:** Inclui variáveis de criação do menu na administração do módulo.
blocks index.html bloque.php	**blocks:** Contém arquivos que definem funções do bloco display/edit. **-blocks.php**: poderá ter o conteúdo do bloco que o módulo possui.
class index.html class.php	**class:** se está programando com orientação a objetos, é conveniente que guarde o código em diferentes arquivos dentro desta pasta. Logo, para usar as classes criadas deve-se somente incluir estes arquivos.
images index.html slogo.png	**images:** diretório onde são armazenadas as imagens. Geralmente, slogo.png é o ícone da administração.
include index.html funciones.inc.php search.inc.php	**include:** neste diretório guarda-se todos os arquivos que contêm funções que se relacionam diretamente com a implementação do módulo. Portanto, para utilizar as funções basta incluí-las nos arquivos que as usam. **-search.inc.php:** corresponde ao arquivo que contém a função de busca do módulo e que será utilizada pelo mecanismo de busca global do XOOPS.
language english index.html admin.php blocks.php main.php modinfo.php spanish index.html admin.php blocks.php main.php modinfo.php index.html	**language:** este diretório contem subpastas que correspondem a uma linguagem que o módulo possui. Estas subpastas contêm os arquivos que definem as constantes de idioma. **-admin.php:** contem a linguagem que será incluída na área de administração. Se o módulo não possuir administração este arquivo não será necessário. **-blocks:** contem a linguagem para os blocos pertencentes ao módulo. Caso contrário, não é necessário manter esse arquivo. **-main.php:** contem a linguagem que será incluída na apresentação do módulo ao usuário. Se o módulo tiver área de administração, este arquivo pode ser omitido. **-modinfo.php:** contem a linguagem que será utilizada principalmente na descrição do sistema. Suas constantes são incluídas na criação do arquivo **xoops_version.php.**
sql index.html mysql.sql	**sql:** neste diretório se encontram os scripts SQL do módulo. Ele possui um arquivo chamado *mysql.sql*, que corresponde aos scripts de criação de tabelas e inserções básicas para o servidor de banco de dados MySQL.
templates index.html theme_index.html	**templates:** Os templates são modelos, HTML principalmente, que contem os dados que o módulo apresentará ao usuário. O módulo pode não ter nenhum template, tudo depende de como o programador deseja criar seu módulo. Caso possua templates, é comum seus arquivos terem nomes associados às páginas que eles pertencem, por exemplo, a página **index.php** possui o template **theme_index.html**
header.php index.php xoops_version.php	**arquivos base do módulo:** local onde se encontram os arquivos que permitirão a instalação do módulo. **-header.php:** A inclusão do núcleo do sistema é feita aqui. **-index.php:** este arquivo dará ao usuário o acesso aos dados do módulo. **-xoops_version.php:** arquivo que contem toda a informação referente ao módulo. Sempre deve existir, pois sem ele o módulo não será instalado.

Quadro 3.3 – Estrutura geral de um módulo XOOPS

De acordo com o quadro 3.3, os principais diretórios presentes na maior parte dos módulos são: {admin/}[18], {blocks/}, {class/}, {images/}, {include/}, {language/}, {sql/}, {templates/} e alguns arquivos base armazenados na raiz do módulo.

Blocos:

Os blocos são partes de um módulo utilizadas para listar informações, na maioria das vezes menus. Todo o módulo XOOPS gera o conteúdo dinâmico para exibir no corpo principal do site, no entanto, muitas vezes é necessário disponibilizar e organizar links em menus, a forma que os módulos utilizam para montar e exibir um menu com opções de ações para o usuário é através dos blocos.

Cada módulo possui uma quantidade específica de blocos, às vezes nenhum. Por exemplo, um módulo de notícias pode oferecer os blocos para listar notícias mais recentes, mais lidas, por autor etc. Todo bloco pode ser configurado na área administrativa do XOOPS para definir em qual divisão da tela (direita, esquerda, centro, rodapé etc) será apresentado e quais atores possuem permissão de visualizá-los.

O XOOPS também permite que os administradores e desenvolvedores criem blocos personalizados, ou seja que não estejam vinculados a nenhum módulo.

Temas:

Um tema define a aparência geral de um site XOOPS. O tema, ou skin, permite a organização do código CSS e HTML necessário para montar a aparência das telas contendo as informações apresentadas aos usuários. Ele define o local onde os blocos ficam localizados e toda a questão gráfica relacionada a apresentação das telas. Como vantagem desta separação está a possibilidade dos usuários trocarem facilmente o tema apresentado, caso não lhe agrade, por um outro.

Da perspectiva do desenvolvedor, um tema consiste em agrupar e colocar os arquivos inerentes a um tema no diretório {themes/}, localizado na raiz do servidor onde o XOOPS está instalado. (Atwal, 2006)

O XOOPS considera todas as pastas contidas no diretório {themes/} como um tema. Com a instalação padrão do XOOPS, são disponibilizados 3 temas pré-definidos, são eles: {default/}, {phpkaox/} e {x2t/}.

Após desenvolver um tema e copiá-lo para o diretório {themes/}, o nome atribuído a

[18] Neste trabalho adotaremos que nomes de arquivos e diretórios e tabelas de banco de dados serão apresentados entre chaves.

pasta torna-se o nome reconhecido pelo XOOPS e que aparecerá no Painel de Controle do Administrador.

Um tema compõe 7 áreas distintas na interface de um site feito em XOOPS, são elas: cabeçalho, coluna esquerda, coluna centro-esquerda, coluna central, coluna centro-direito, coluna direita e rodapé. A figura 3.20 mostra como é dada esta divisão.
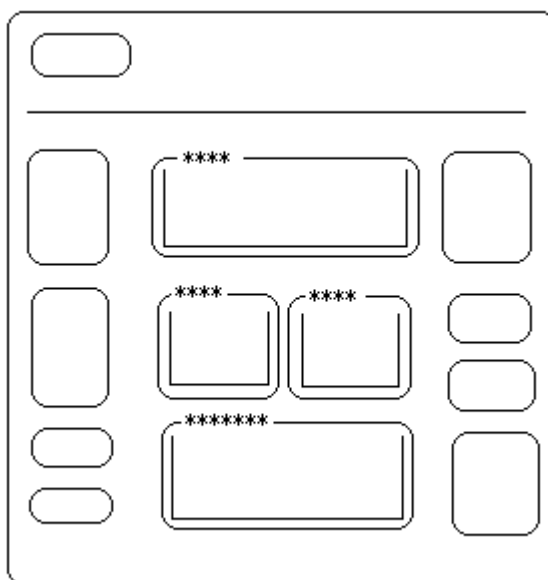
Figura 3.20 – Divisão esquemática das áreas de conteúdo do tema do XOOPS

O arquivo mais importante que compõe um tema é um arquivo HTML denominado {theme.html}. Ele organiza integralmente o tema, ou seja, define os tamanhos e posições de cada uma das 7 áreas, folha de estilo etc. Excetuando-se o cabeçalho e o rodapé, cada área gera um bloco de código, necessário a sua implementação, que pode ser escrito dentro do próprio {theme.html} ou em arquivos individuais, fazendo-se as devidas referências dentro do {theme.html}.

3.2.5 – O XOOPS no Desenvolvimento do EVA

O paradigma híbrido adotado para construção do sistema, o modelo em cascata baseado em reuso de componentes, sugere a adoção do reuso de componentes na etapa de codificação, desta forma foi realizado um levantamento de módulos que implementassem as funcionalidades modeladas na etapa de projeto ou que mais se aproximassem. A procura foi concentrada principalmente no repositório de módulos da comunidade oficial do XOOPS[19]. As buscas foram realizadas também nas comunidades virtuais sobre XOOPS dos principais países envolvidos no projeto XOOPS, como por exemplo, a comunidade italiana

[19] O Site oficial do XOOPS, conhecido como XOOPS Project está acessível no endereço eletrônico www.xoops.org.

(www.xoopsitalia.org), a comunidade mexicana (www.xoopsmexico.net) e as nacionais[20].

O levantamento por módulos nas comunidades XOOPS teve como objetivos módulos que relacionassem com os destinados ao desenvolvimento inicial, assim o levantamento de módulos já desenvolvidos consistiu em buscar módulos com características de um controlador (principal), fórum, biblioteca de resenhas, avisos, chat, e Estudos de Caso. A seguir é apresentada uma breve descrição de cada um dos módulos selecionados durante a busca e qual a funcionalidade que ele implementa dentro da estrutura. São apresentadas, também brevemente, as modificações realizadas no módulo para se encaixar na especificidade do sistema.

Principal

O SGC XOOPS, como já apresentado, possui um módulo central de controle denominado [system]. O módulo [system] possui as mesmas características modeladas para o módulo principal, de permitir a gerencia de grupos, usuários, módulos etc (figura 3.21).

20 São três as comunidades nacionais de suporte ao XOOPS: www.xoopsbr.org, www.xoopstotal.com.br e www.xoops.pr.gov.br.
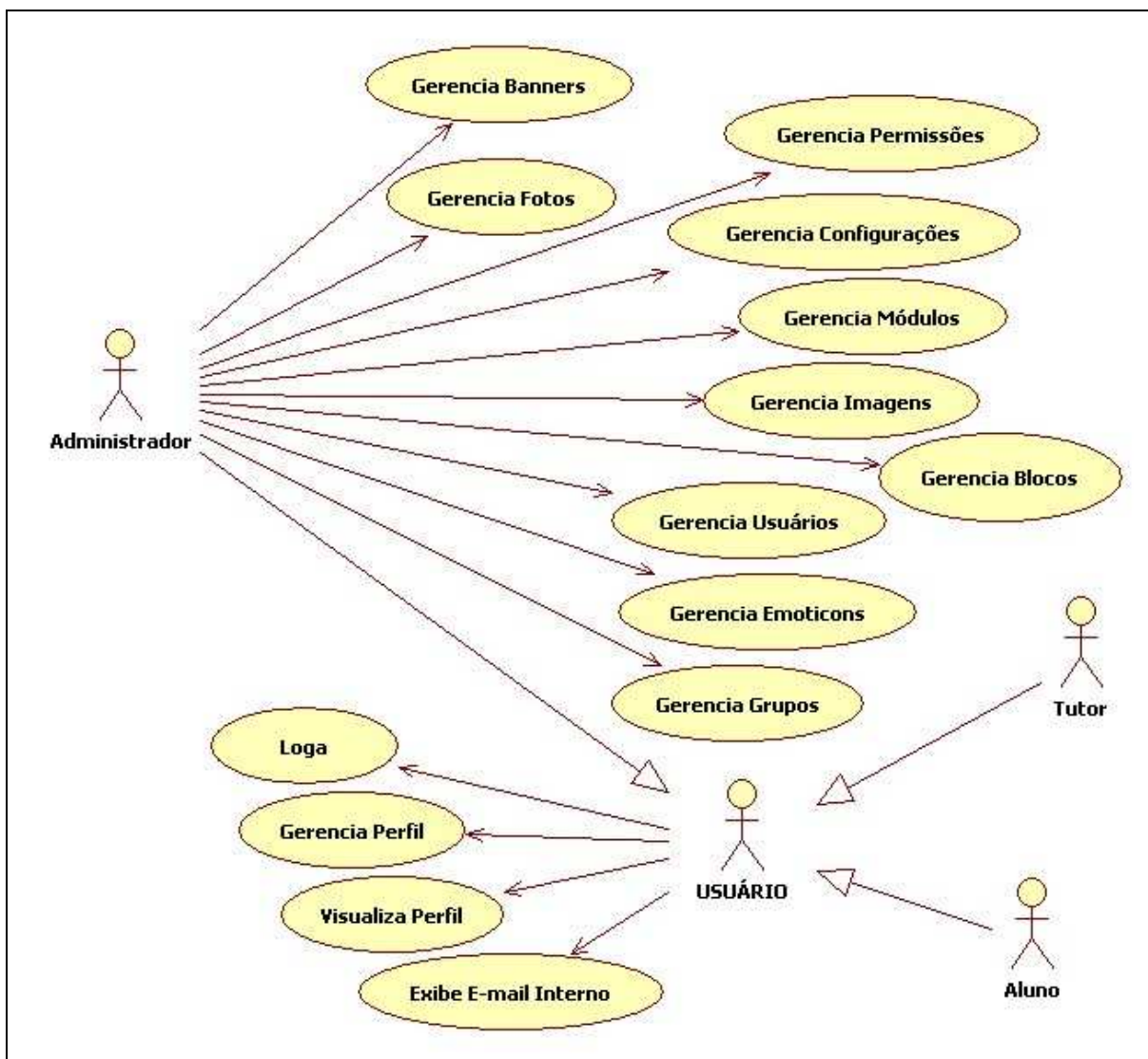
Figura 3.21 – Diagrama de casos de uso do módulo [system] do XOOPS

Quando comparado o diagrama de casos de uso do módulo [system] do XOOPS (Figura 3.21) com o principal modelado no início do projeto (Figura 3.6), percebe-se que todos os casos de uso do módulo principal, nas visões do Professor, Aluno e Administrador, já estão desenvolvidos no módulo [system]. Assim este módulo atendeu a necessidade do projeto de um módulo central controlador da arquitetura dos módulos na camada de lógica do negócio.

Nenhuma adaptação estrutural foi realizada neste módulo, já que não houve a necessidade de desenvolver casos de uso não atendidos ou atendidos parcialmente. As únicas alterações consistiram em revisar a tradução.

Foi necessário amplo estudo do funcionamento do módulo, no sentido de compreender seus padrões lógicos de relacionamento, a fim de permitir a modelagem e implementação de novos recursos sem corromper a estrutura do SGC.

Fórum

O levantamento por módulos com características de fórum foi facilitada por haver diversos projetos de desenvolvimento de módulos de fórum, desta forma o objetivo foi selecionar o que mais se aproximasse da modelagem inicial estabelecida. Neste sentido optou-se pelo módulo denominado newbb. O [newbb] foi o primeiro módulo desenvolvido com esta característica para o XOOPS, sua arquitetura é baseada no sistema de fórum [phpbb]. Atualmente o [newbb] está traduzido para 10 idiomas inclusive o português do Brasil. Seu desenvolvimento se deu através do trabalho colaborativo entre desenvolvedores e comunidades XOOPS do Canadá, Estados Unidos e China.

O newbb possui, em virtude do longo período de desenvolvimento e testes do módulo em diversas comunidades, grande quantidade de recursos e funcionalidades. Porém a maior parte deles se relaciona com a camada de apresentação tornando possível a exibição e ordenação do conteúdo de diversas formas.

A camada lógica do [newbb], relacionada com a modelagem inicial, implementa funcionalidades que contemplam as estabelecidas na modelagem inicial. Na figura 3.22 é apresentado um diagrama de casos de uso, que apresenta a estrutura do módulo [newbb].
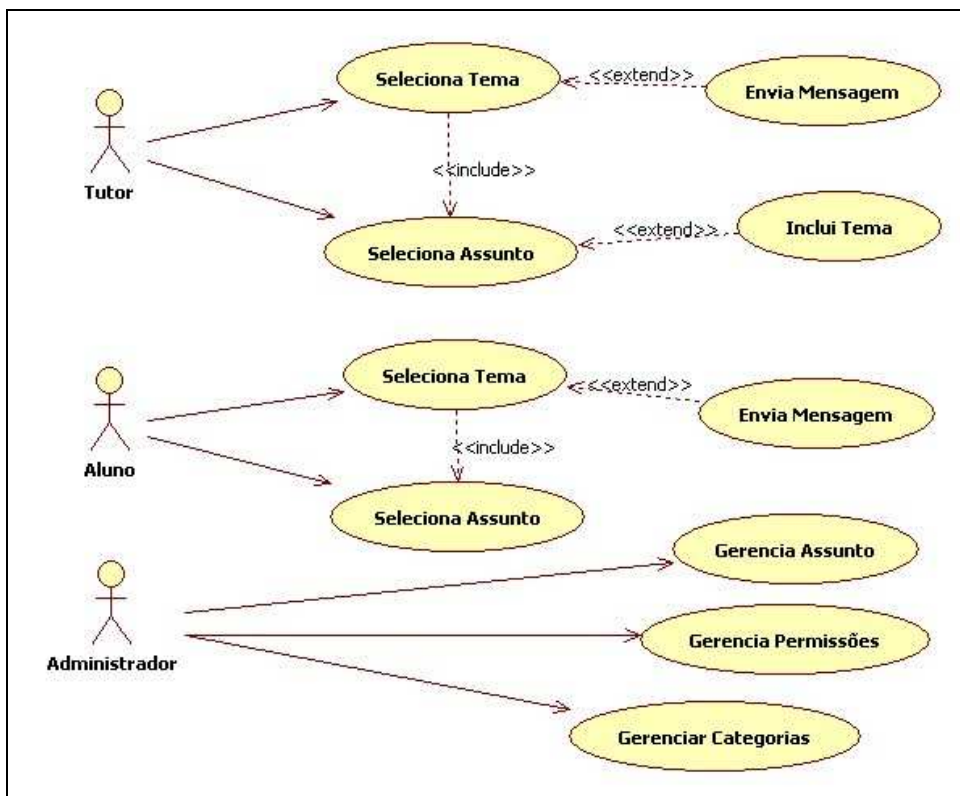
Figura 3.22 – Diagrama de casos de uso do módulo newbb

Observa-se na estrutura do módulo [newbb] que ele implementa todos os casos de uso modelados na fase de projeto para o fórum.

Um ponto de adaptação importante foi acerca das permissões, no módulo [newbb] todos os assuntos são vinculados a uma categoria. O módulo permite que se atribua permissões de acesso as categorias de acordo com os grupos existentes. Desta forma solucionou-se a questão de permissões da seguinte forma: ao se criar um grupo, cria-se automaticamente uma categoria no fórum com exatamente o mesmo nome do grupo. Cada Estudo de Caso associado a este grupo, implica em se criar automaticamente um assunto no fórum, inserido na categoria referente ao grupo. A partir disso o professor poderá abrir temas neste assunto do fórum, pois estará indicando que ele pertence a um assunto de um Estudo de Caso de um grupo.

Estas adaptações foram necessárias para adequar o módulo às características pretendidas sem violar a estrutura do módulo [newbb]. Outras adaptações realizadas foram de caráter da apresentação, ou seja, reorganizar aspectos da interface.

Biblioteca de Resenhas

A biblioteca de resenhas consiste em uma seção do ambiente em que os professores podem armazenar resenhas de leituras indicadas e demais materiais relacionados ao Estudo de Caso. Esta seção baseia-se nos conceitos da biblioteca de resenhas do AVEC, ou seja, o professor insere os materiais, que são disponibilizados no formato de uma lista, a partir daí os professores podem sugerir que os alunos efetuem o download do material.

Para implementação desta seção, buscou-se algum módulo com características de repositório de arquivos. A procura por módulos revelou a existência de vários módulos destinados ao armazenamento de arquivos, que são conhecidos como módulos de download.

O módulo mais próximo dos nossos requisitos foi o módulo [wf-download] da equipe de desenvolvimento WF-Project team. Este módulo possui como característica permitir que um usuário, com permissões suficientes, preencha os dados de um determinado arquivo, realize seu upload e submeta-o ao banco de dados. Considera-se também que o usuário deverá associar o arquivo a alguma categoria criada pelo administrador.

Os arquivos são apresentados em uma lista de materiais divididas por categoria sem divisão de grupos, todos os usuários podem visualizar todos os materiais, porém recomenda-se que siga as orientações de leituras dos professores.

Neste primeiro momento de codificação não houve necessidade de adaptações no módulo, apenas algumas correções de tradução. Utilizou-se o banco de dados e telas do módulo original [wf-download].

Avisos

O levantamento realizado com o objetivo de encontrar módulos com características de um módulo de avisos revelou que não existia nenhum módulo com essa especificidade nas comunidades XOOPS consultadas.

Um novo levantamento foi realizado, desta vez com o objetivo de localizar um módulo com características aproximadas de um módulo de avisos, que fosse passível de adaptação. Neste contexto, o módulo selecionado foi um módulo de notícias desenvolvido pelo brasileiro Gustavo S. Villa. O módulo denominado [eNoticias] possui uma estrutura direcionada à publicação de notícias, como apresentado na figura 3.23, no entanto mostrou-se adaptável.
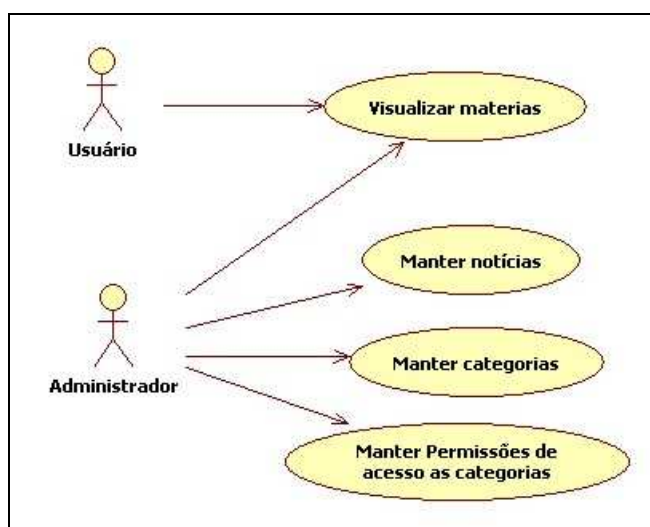
Figura 3.23 – Casos de uso do módulo [eNoticias]

Para persistência dos dados, o módulo [eNoticias] utiliza três tabelas de banco de dados: {enoticias_artigos} (permite o armazenamento das categorias), {enoticias_colunas} (permite o armazenamento das notícias) e {enoticias_votacoes} (permite o armazenamento das avaliações das notícias pelos usuários).

A estrutura deste módulo de notícia apresenta uma coleção de notícias relacionada a uma categoria principal. Esta estrutura assemelha-se ao modelo de módulo de avisos, definido na fase de projeto, em que houve uma coleção de avisos relacionada a um grupo.

A mudança conceitual do módulo foi realizada através de renomeação de tabelas no banco de dados e reconstrução de toda a interface gráfica. O primeiro relacionado à padronização de sintaxe e o segundo, à forma de apresentação do conteúdo de maneira adequada as definições do módulo.

Uma adaptação importante no módulo está relacionada às permissões. O módulo

[eNoticias] permite que se atribua permissões de acesso as categorias de acordo com os grupos existentes. Desta forma solucionou-se a questão de permissões da seguinte forma: ao se criar um grupo, cria-se automaticamente uma categoria no módulo com exatamente o mesmo nome do grupo. Assim é possível o administrador aplicar a permissão para um grupo acessar apenas os avisos contidos na categoria relacionada ao seu grupo de estudo.

Chat

Umas das ferramentas de comunicação mais importantes do sistema didático EVA. O chat é uma função do sistema presente no AVEC e que optou-se por mantê-lo no EVA. O levantamento por módulos de chat revelou que esta é uma categoria de módulos em que o XOOPS não possui grande variedade de bons módulos. A maior parte deles estava em fase inicial de desenvolvimento ou utilizam sistemas em flash que destoam da estrutura padrão dos demais módulos. Desta maneira selecionou-se o módulo que mais se aproximou da estrutura especificada. O módulo selecionado foi o [khat], desenvolvido pela comunidade italiana do XOOPS.

Apesar do módulo [khat] apresentar arquitetura simples, ele se aproximou mais das necessidades, pois atualiza mais rapidamente a tela com as mensagens, refletindo a vantagem do chat frente ao fórum, ser uma ferramenta para diálogo síncrono. Permite a inserção de emoticons.

O módulo [khat] não possui uma das principais funções esperadas para o módulo de chat, permitir a criação de salas. Este recurso não foi encontrado em nenhum dos demais módulos testados, no entanto o [khat] possuía a arquitetura mais flexível à adaptação. O módulo possui um banco de dados modelado para permitir a criação de salas de discussão, necessitava as devidas complementações de código para adequá-lo.

Assim foi realizado estudo do módulo e seus códigos a fim de viabilizar as alterações necessárias. Verificou-se que o módulo possuía todos os seus códigos direcionados a associação das mensagens com uma sala de bate-papo específica, como a funcionalidade de criação de salas não estava desenvolvida, ele utilizava como padrão o código ‘0’ para identificar a sala das mensagens. Nesta estrutura todas as mensagens pertenciam a mesma sala de bate-papo, a sala ‘0’.

A modificação consistiu em fazer com que o módulo reconhecesse o código de um grupo e o utilizasse para identificar as mensagens. Por exemplo, se um grupo possui como código o número ‘5’, todas as mensagens enviadas na sala de bate-papo deste grupo serão associadas ao código ‘5’, não mais ao ‘0’.

Apesar do módulo não possuir uma interface própria para criação de salas no chat, ele consegue distinguir quais as mensagens pertencem a um grupo através do código deste. A tela inicial do módulo na visão de um grupo é apresentada na figura 3.24.
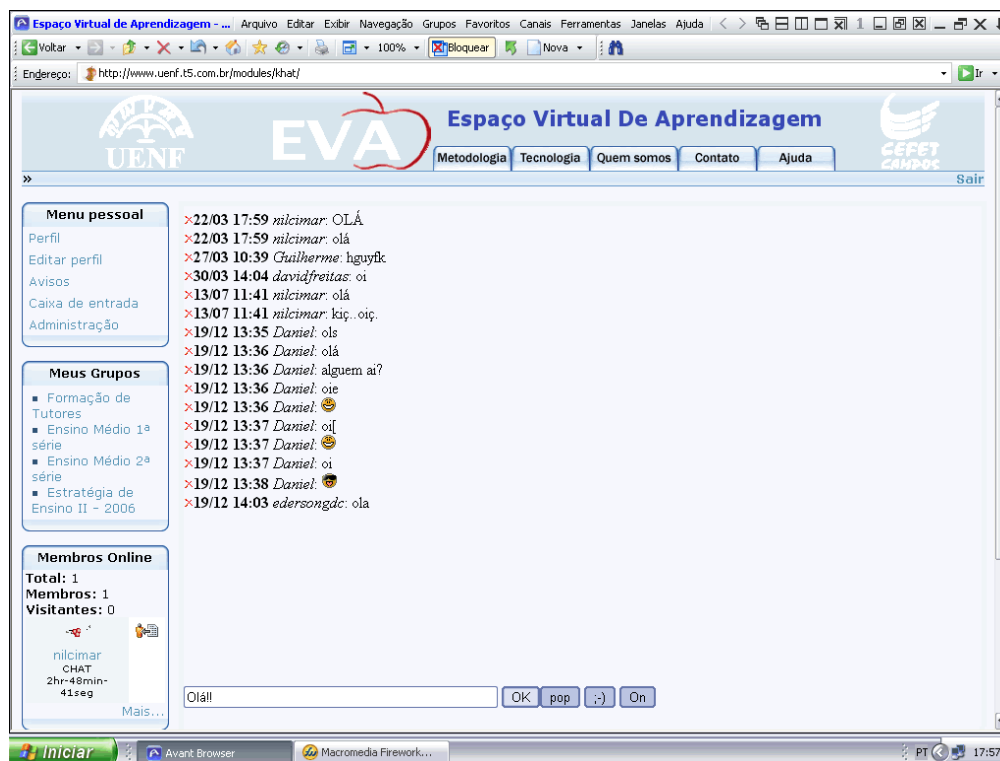
Figura 3.24 – Tela inicial do módulo de chat

As mensagens são listadas na tela precedidas da data e horário de envio e do nome do usuário que a enviou. Na visão do administrador há ainda um "x" em vermelho no início da linha, que possibilita a exclusão da mensagem.

Estudos de Caso

O levantamento por módulos que implementassem o modelo conceitual de ABC revelaram a inexistência de um módulo desenvolvido, ou mesmo parcialmente desenvolvido, com esta especificidade. Também se verificou que nenhum módulo, dentre os consultados, se mostrou adaptável ao modelo pretendido. Portanto o módulo de Estudos de Caso teve de ser desenvolvido integralmente pela equipe do projeto, implicando amplo estudo da arquitetura do XOOPS, dos módulos, blocos e lógicas de funcionamento.

Os casos de uso definidos para este módulo foram:

- Visualizar Estudos de Caso (acessível à todos)
- Visualizar grupos de estudo (acessível à todos)
- Corrigir passos (acessível aos professores)

- Manter grupos de estudo (acessível aos administradores)
- Manter Estudos de Caso (acessível aos administradores)
- Associar Estudo de Caso ao grupo (acessível aos professores)
- Responder passos (acessível aos alunos)

Primeiramente definiu-se alguns padrões de apresentação. Foi definido que a visualização dos grupos de estudo se daria através de um bloco. O bloco conteria todos os grupos de estudo ativos do usuário. Após selecionar um dos grupos, o usuário seria redirecionado a uma tela contendo os Estudos de Caso referentes ao grupo. Esta tela daria acesso também as ferramentas de chat, cronograma, avisos, fórum e repositório de arquivos. Ao selecionar uns dos Estudos de caso listados, o usuário veria uma tela com o texto do Estudo de Caso, e um link para a correção de estudos, na visão do professor ou um link para responder os passos, na visão do aluno. As ações de manter Estudos de Caso e grupos de estudo ficariam restritas aos administradores na interface administrativa do XOOPS.

Criou-se, com base na estrutura padrão dos módulos XOOPS, o módulo de Estudo de Caso, com o nome de [evax_admin]. Para exibição dos grupos foi criado um bloco com o título de "Meus Grupos", nele o usuário ao acessar o ambiente pode visualizar os grupos a que ele pertence (figura 3.25).

Figura 3.25 – Bloco "Meus Grupos" do módulo de Estudos de Caso

Para a exibição dos Estudos de Caso foi criada uma tela que exibe em forma de lista todos os Estudos de Caso de um grupo (figura 3.26).
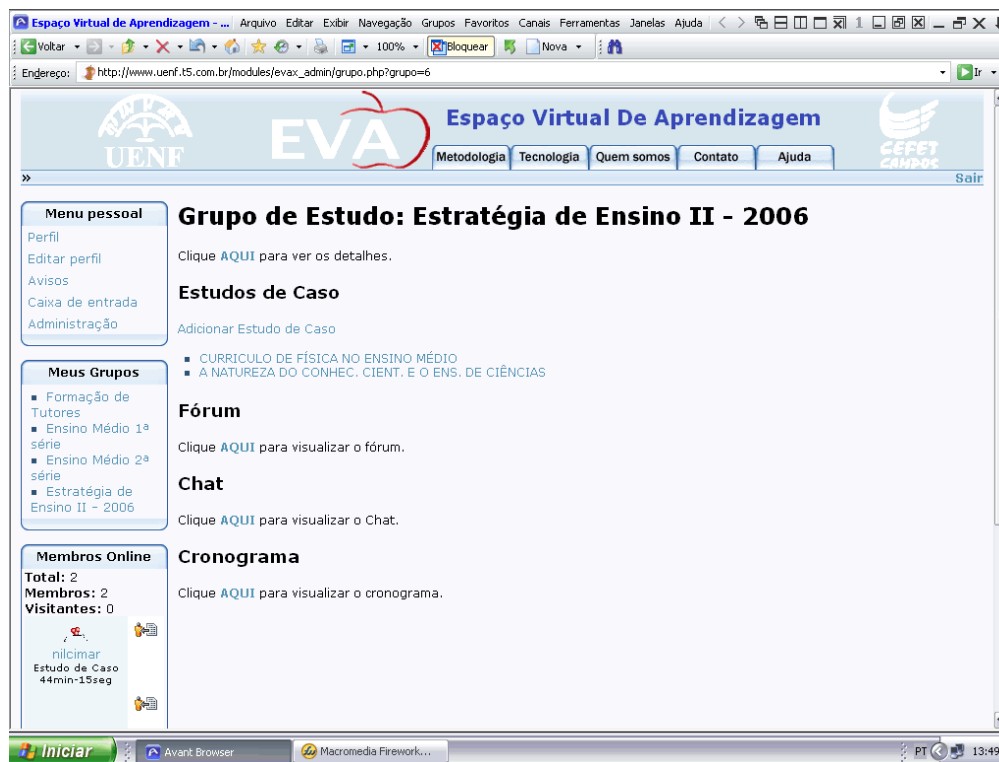
Figura 3.26 – Tela inicial do módulo de Estudos de Caso

Nesta tela foram inseridos outros links, que deram acesso as ferramentas de fórum, chat e cronograma. Esta tela pode ser acessada por qualquer membro de um determinado grupo, porém não se permite o acesso por usuários de outros grupos.

Ao associar um Estudo de Caso a um grupo, criam-se automaticamente os quatro passos referentes àquele Estudo de Caso. O acesso às telas de resposta e correções de passos dá-se através da tela inicial de um Estudo de Caso (figura 3.27).
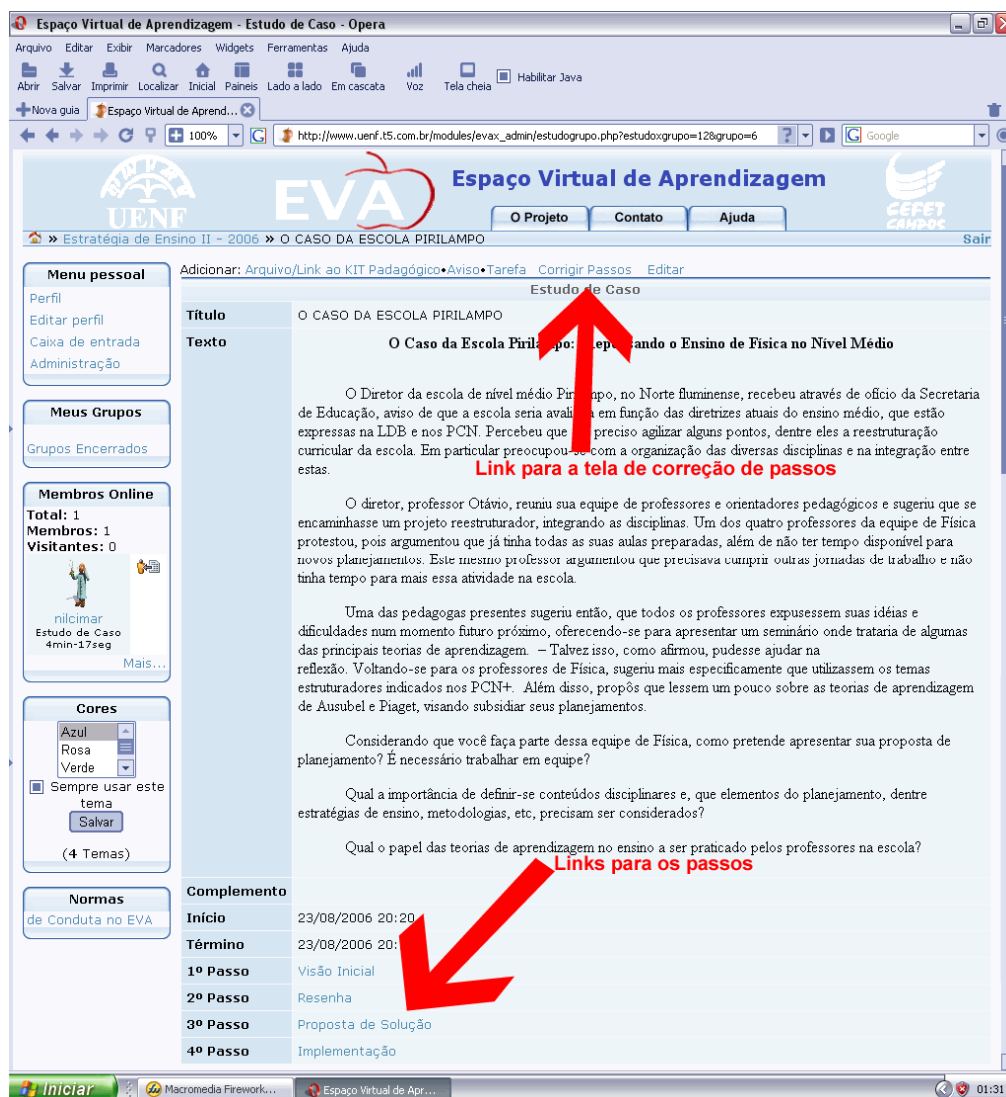
Figura 3.27 – Tela inicial do módulo de Estudos de Caso

Na parte superior o link representado pela expressão “corrigir passo” permite o professor acessar a tela de correção dos passos dos estudantes (figura 3.28). Na parte inferior destaca-se os links para os quatro passos criados.

De acordo com a estratégia de Estudos de Caso adotada no EVA (Reis, 2008), no passo 1 o estudante deve apresentar uma resposta que exponha sua visão original (concepção existente) sobre a questão orientadora do Estudo de Caso. No passo 2, é necessário resenhar o texto de base indicado. No passo 3 o estudante deve apresentar uma solução com defesa das principais idéias. O passo 4, de caráter opcional, pode ser solicitado, desde que o estudo encaminhe implementações.
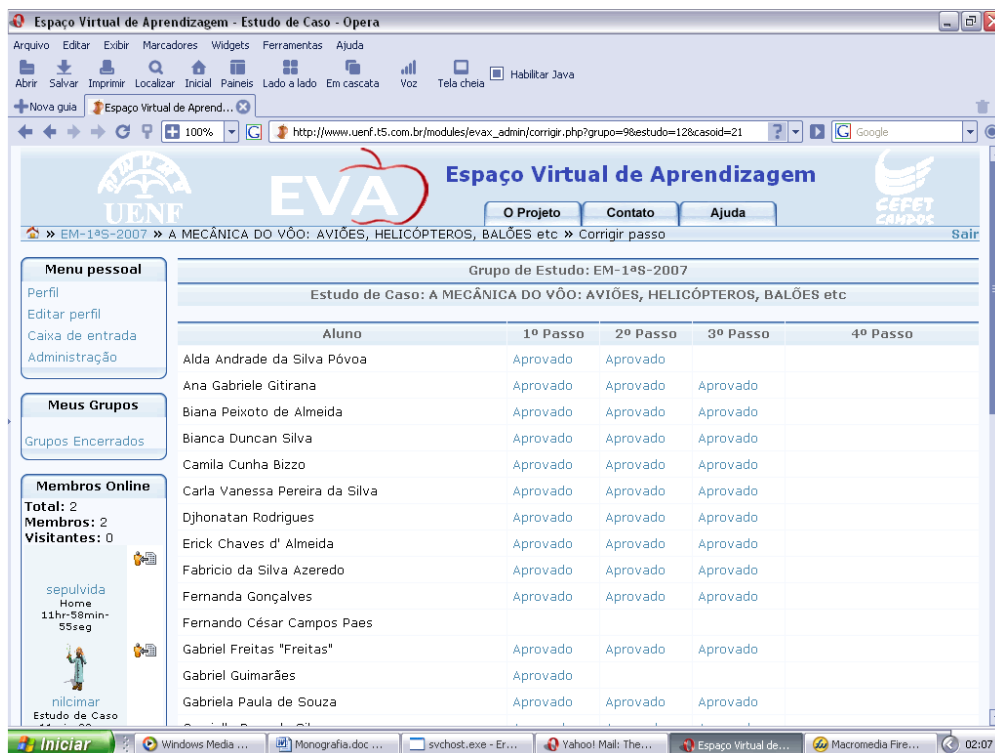
Figura 3.28 – Tela de correção dos passos

A avaliação do professor fica disponível em uma tela em que o aluno pode enviar sua resposta, verificar a avaliação do professor e refazer sua resposta, quando solicitado.

A figura 3.29 mostra a tela de um passo, na qual o aluno observa a avaliação do professor.
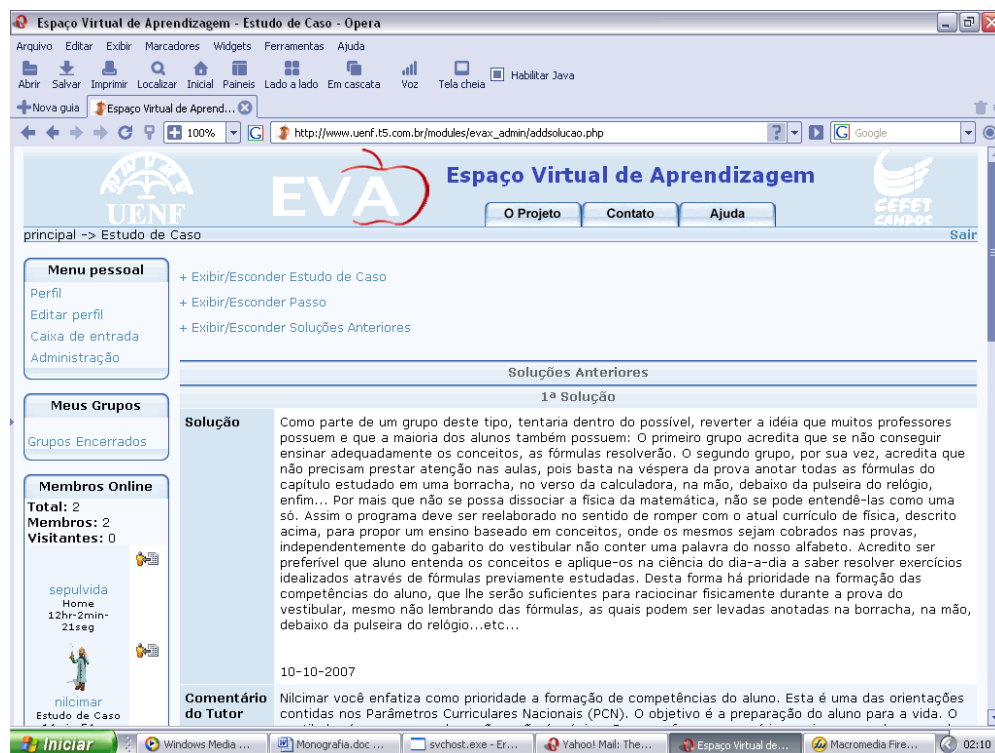
Figura 3.29 – Tela de correção dos passos

Todos os requisitos deste módulo apresentadas até aqui se relacionam com a organização, exibição e controle do conteúdo. As partes que envolvem a estrutura do sistema e sua interação com o XOOPS estão presentes nas visões dos administradores. Para a criação de grupos de estudo, utilizou-se a ferramenta de gerencia de grupos do XOOPS, ela permite que se apliquem permissões diferenciadas aos grupos, que consequentemente serão herdadas pelos usuários do grupo.

Para implementar permissões distintas aos professores e alunos de um grupo explorou-se a ferramenta de gerencia de grupos do XOOPS da seguinte forma: ao criar um grupo no módulo [evax_admin], automaticamente são criados dois grupos no XOOPS, com as extensões "_TUTOR" e "_ALUNO", por exemplo, ao criar o grupo "Ensino Médio 2008" no módulo [evax_admin], ele criará automaticamente os grupos "Ensino Médio 2008_TUTOR" e "Ensino Médio 2008_ALUNO" no XOOPS, desta forma é possível atribuir permissões distintas à professores e alunos de um mesmo grupo de estudo.

A associação de um Estudo de Caso a um grupo é efetuada em dois momentos, no primeiro o administrador deve cadastrá-lo no banco de dados, em seguida o professor poderá associá-lo ao grupo. Esta decisão possibilita que um Estudo de Caso seja adicionado ao banco de dados uma só vez.

A tela em que o administrador têm as opções de gerenciar os grupos de estudo e os Estudos de Caso é apresentada na figura 3.30.
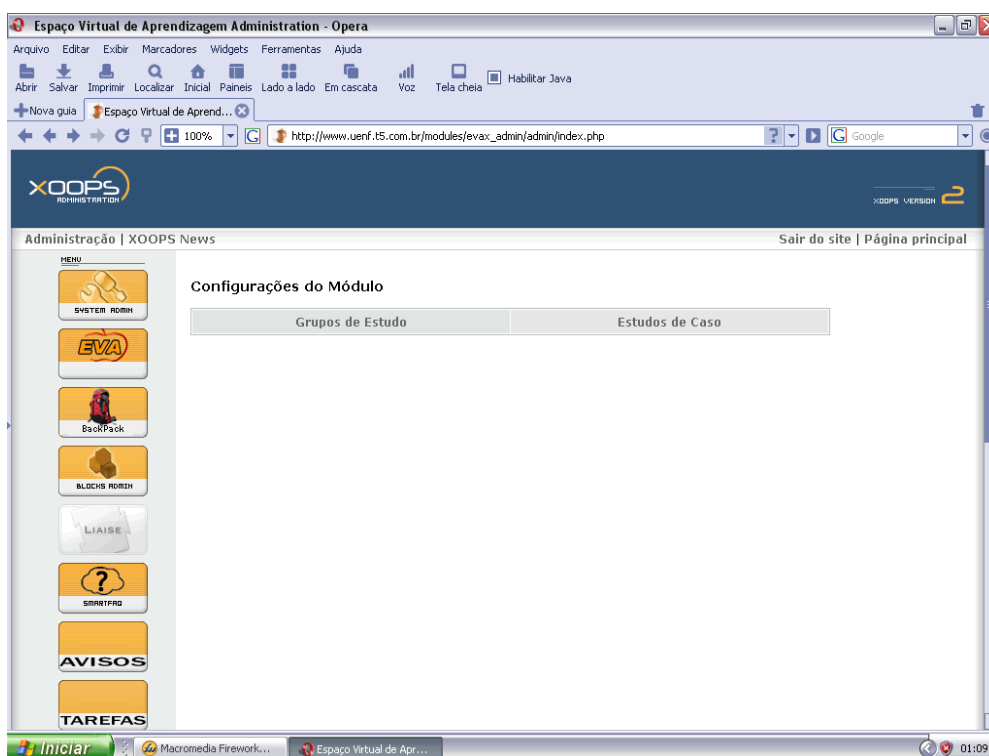
Figura 3.30 – Tela administrativa do módulo de Estudos de Caso

Na área principal da tela apresenta-se o link para a tela de gerência de grupos de estudo (à esquerda) e o link para a tela de gerência de Estudos de Caso (à direita).

3.2.6 – Pré-testagem

A pré-testagem do sistema ocorreu nos anos de 2005 e 2006 em 3 turmas da Licenciatura, na disciplina de Estratégia para o Ensino de Física I. II e III e em 2 grupos de estudantes do primeiro e segundo anos do Ensino Médio, na disciplina curricular de Física.

Parte fundamental no paradigma clássico, o pré-teste é realizado com o objetivo de ser o primeiro grande teste do sistema e a partir dele se estabelecer novos requisitos e novo escopo para a seqüência de desenvolvimento.

O pré-teste com estudantes do Ensino Médio consistiu em uma experiência didática com 59 estudantes, sendo 28 da segunda série e 31 da primeira série. Foram abertos dois grupos de estudo, cada relacionado a uma série. A experiência foi realizada ao longo de um bimestre letivo. Um professor de Física, com especialização em Informática foi responsável pela orientação do Estudo de Caso, que complementou ações presenciais de suas aulas de Física.

O foco informático da experiência didática foi investigar a capacidade do sistema em suportar a aplicação do modelo didático ABC a grupos de estudo reais, fora do ambiente de desenvolvimento.

Após esta primeira implementação, a unidade conceitual e o próprio EVA passaram por uma reestruturação, baseada em observações da equipe e nas sugestões e críticas apresentadas pelos alunos no decorrer do processo.

3.2.7 - Remodelagem de Escopo

O pré-teste foi usado, como sugere o modelo em cascata, para que fosse feita uma avaliação do sistema e das ferramentas implementadas. Nesta etapa, os temas de discussão foram a codificação de novas funcionalidades e a reformulação e/ou otimização das desenvolvidas.

A partir do momento em que o sistema entrou em operação, os novos requisitos surgiram e tornou-se necessário outras remodelagens. A avaliação realizada nesta etapa se caracteriza de três formas diferentes: avaliação dos módulos não priorizados na fase anterior, avaliação dos módulos desenvolvidos e avaliação da necessidade de novos módulos.

Avaliação dos módulos não priorizados na fase anterior

Não se priorizou no sistema inicial os módulos: mural, catálogo de usuário, repositório de arquivos e portfólio. Nesta etapa de remodelagem de escopo e instrucional do ambiente, verificou-se que o módulo mural não era necessário no modelo, visto que o módulo de avisos cumpria, também, está função. O módulo catálogo de usuário, assim como o módulo mural, foi considerado desnecessário, já que o módulo de Estudos de Caso lista os alunos e professores de cada grupo. O módulo repositório de arquivos e o módulo biblioteca de resenhas (já desenvolvido) foram unidos em um único conceito de kit pedagógico. O portfólio não foi considerado para codificação na versão beta, porém não foi descartado e sim adiado para momento em que haja pesquisa mais consistente acerca do tema.

Avaliação dos módulos desenvolvidos

Esta Avaliação revelou a necessidade de criar novos recursos em dois módulos desenvolvidos, o [system] do núcleo XOOPS e o de Estudo de Caso [evax_admin]. No módulo [system] do XOOPS detectou-se a necessidade do recurso "Onde Estou?", que consiste em uma linha na parte superior das telas com o caminho percorrido pelo usuário até chegar àquela tela. No entanto a maior necessidade de acréscimo de requisitos ocorreu no módulo de Estudos de Caso, são eles: Nova tela de acesso às funcionalidades do EVA apresentada de acordo com as visões permitidas ao usuário que a acessa, Gerência de grupos e Ferramenta de gerencia do professor, que se subdivide em relatórios, estatísticas, gerência de conteúdo e bibliotecas de materiais pedagógicos e Estudos de Caso.

Avaliação da necessidade de novos módulos

As necessidades de inserir ferramentas não foram restritas a criação de novos requisitos nos módulos já desenvolvidos, houve também a necessidade de inserção de novos módulos, são eles:

- Ajuda – Ferramenta que visa solucionar dúvidas do aluno quanto à navegação no ambiente. Consiste de arquivos que demonstram as telas do ambiente com suas ações e recursos.
- Backup – Ferramenta de segurança de uso restrito da administração, acionada periodicamente para garantir a possibilidade de recuperação de dados.
- Tarefas – Este espaço pode ser utilizado para direcionar as ações pedagógicas ou enviar materiais de forma mais ágil e explicitada.

- Texto estático – Armazena e apresenta todo o conteúdo estático produzido.
- Cronograma – Ferramenta auxiliar na organização do grupo.
- Contato – Espaço utilizado pelos usuários para enviar qualquer tipo de informação aos e-mails dos administradores, desde sugestões até dúvidas que não tenham sido esclarecidas pela ajuda.
- Personalização da interface – Possibilidade do usuário alterar o as cores da interface.
- Kit pedagógico – Repositório de material didático. Cada estudo de caso possui seu kit pedagógico.
- Usuários online: Executa ação de indicar os logins ativos.
- Estatísticas – Executa ações relacionadas à pesquisa sobre as ações dos estudantes e dos grupos – pode ser utilizada para gerar gráficos, tabelas etc.

A próxima seção trata o desenvolvimento das funcionalidades abordadas na etapa de remodelagem de escopo em uma nova versão do sistema, denominada de versão beta.

3.3 – Versão Beta

Para implementar os requisitos surgidos na etapa de remodelagem verificou-se a necessidade de maior interação entre os desenvolvedores e os usuários, já que o modelo em cascata proporciona esta interação somente durante a etapa de análise de requisitos. Esta interatividade foi obtida com o desenvolvimento de uma versão beta de testes do sistema.

De acordo com Pressman (2007),

> *"O teste beta se caracteriza por ser uma aplicação "viva" do software, num ambiente que não pode ser controlado pelo desenvolvedor. O cliente registra todos os problemas (reais ou imaginários) que são encontrados durante o teste beta e relata-os ao desenvolvedor a intervalo regulares"* (p.861).

O paradigma de engenharia de software adotado para o desenvolvimento inicial do EVA, o modelo em cascata, não comportaria esta nova fase de desenvolvimento. Havia necessidade de um modelo que possibilite alterar um requisito de forma mais rápida.

O modelo em cascata atendeu as necessidades iniciais, mas tornou-se inadequado à seqüência do projeto, pois a implementação de uma quantidade alta de requisitos, que podem ser alterados em um sistema já iniciado, necessita de um paradigma de desenvolvimento que proporcione maior dinamismo na relação desenvolvedor-cliente ao longo de todo o processo e não apenas na fase de análise. Esta afirmação percebe-se na observação de Pressman (2007), em que: "*a engenharia de software compõe-se, dentre outras coisas, pela combinação de*

paradigmas em todas as fases do desenvolvimento” (p.30).

É possível retomar os modelos preteridos inicialmente: prototipal ou evolutivo e o espiral. Dentre estes modelos, optou-se por utilizar no desenvolvimento da versão beta o modelo prototipal, também conhecido como modelo evolucionário, pois se apresenta como a “*melhor abordagem quando durante a elaboração de um projeto define-se um conjunto de objetivos gerais a serem alcançados e não se tem informação detalhada dos requisitos*” (Sommerville, 2007, p.45).

No EVA os desenvolvedores recebem a especificação dos requisitos, porém esta não é apresentada de forma definitiva, poderá ser alterada após o sistema entrar em operação. O modelo evolutivo se mostrou mais adequado à seqüência de desenvolvimento, pois o paradigma permitiu a especificação de requisitos de forma incremental, os usuários compreendem melhor seus problemas.

A abordagem mais adequada do modelo no contexto do EVA foi uma extensão do significado de evolucionário, para evolucionário exploratório, em que “*o objetivo do processo é trabalhar com o cliente a fim de explorar seus requisitos até alcançar um nível de desenvolvimento adequado e satisfatório*” (figura 3.31) (Sommerville, 2007). As características deste modelo representam a interatividade entre desenvolvedores e clientes pretendida para esta nova fase de desenvolvimento do EVA. Desta forma este paradigma foi adotado para a seqüência de desenvolvimento do EVA.
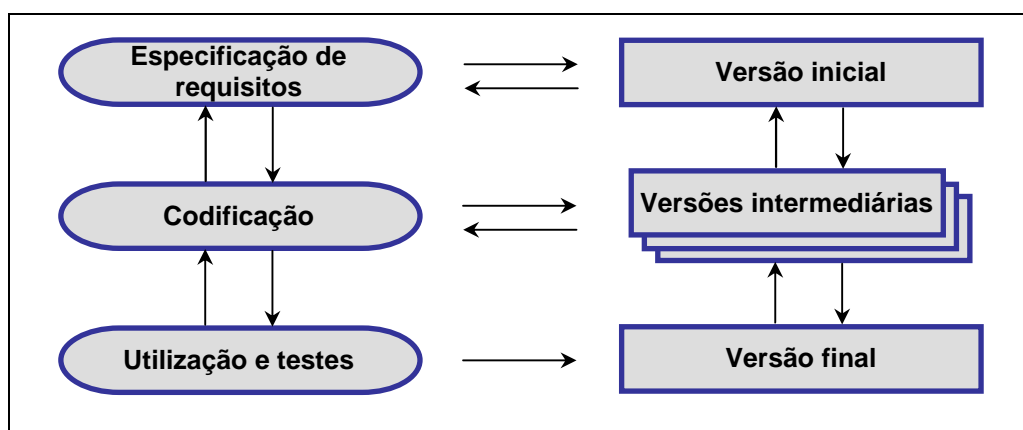
Figura 3.31 – Paradigma de desenvolvimento evolucionário exploratório

Após a opção de utilização do modelo evolucionário exploratório, foi necessário adequar o ciclo de vida do sistema, que era o modelo em cascata baseado em componentes. A adoção do paradigma evolucionário em substituição ao modelo em cascata não excluí a utilização do paradigma auxiliar, baseado em componentes, pelo contrário, Sommerville (2007) afirma que “*na abordagem evolucionária o reuso é freqüentemente essencial para o*

desenvolvimento rápido do sistema e que a utilização formal do reuso pode ser obtida através do paradigma baseado em componentes" (p.46).

A seqüência de desenvolvimento da versão beta do sistema seguiu o paradigma evolucionário exploratório baseado em componentes.

Para combinar o paradigma evolucionário (figura 3.31) com o baseado em componentes (figura 3.18) é necessário depreender que os dois possuem seu início na especificação de requisitos e fim na Utilização e Testes do sistema.

O paradigma evolucionário propõe etapas para as versões produzidas: desenvolver uma versão inicial, várias intermediárias e uma final, enquanto o paradigma baseado em componentes descreve uma série de etapas para a codificação, assim a seqüência intermediária do paradigma baseado em componentes: análise de componentes → modificação de requisitos → projeto de sistema com reuso → codificação e integração, pode substituir sem prejuízo a etapa de codificação do paradigma evolucionário, resultando no modelo combinado, híbrido apresentado na figura 3.32.
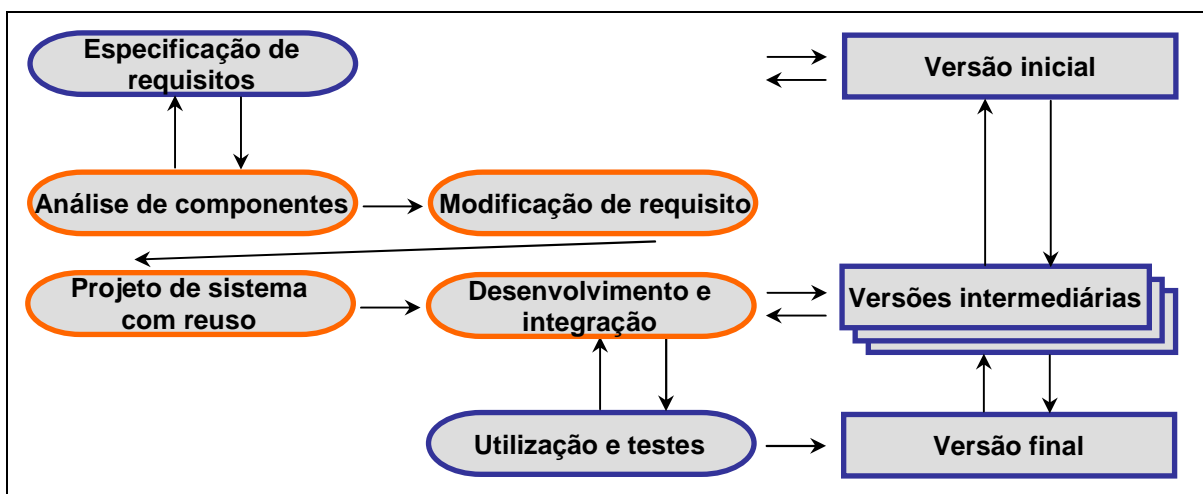
Figura 3.32 – Paradigma híbrido – evolucionário exploratório baseado em componentes

Para o desenvolvimento da versão beta teste do sistema denominou-se as etapas como i) Especificação de requisitos, ii) Projeto, iii) Codificação e iv) Utilização.

3.3.1 – Especificação de Requisitos

A especificação dos requisitos sofreu forte influência da utilização das versões betas. No período entre a remodelagem de escopo e a utilização, o EVA foi testado em 3 turmas do curso de licenciatura em física da UENF nas disciplinas de estratégia I, II e III (Reis, 2008). A interação entre os desenvolvedores e usuários foi obtida através de oficinas e reuniões. No escopo desta monografia serão definidos apenas os requisitos desenvolvidos e mantidos para a

versão utilizada na utilização com estudantes de Ensino Médio. Portanto, não há descrições de todas as alterações ocorridas entre as versões intermediárias testadas.

No quadro 3.4 são apresentados os requisitos funcionais que se espera disponibilizados na versão utilizada no Ensino Médio. Estes requisitos refletem as especificações de novos módulos e adaptações nos existentes descritas na remodelagem de escopo.

Ator	Requisito Funcional
Professor	• Deve ser capaz de publicar uma tarefa. • Deve ser capaz de armazenar e gerenciar um material na biblioteca de materiais didáticos e associá-lo ao kit pedagógico de um de seus grupos. • Deve ser capaz de armazenar e gerenciar um Estudo de Caso na biblioteca de Estudo de Casos e associá-lo a um de seus grupos. • Deve ser capaz de gerenciar o cronograma de seu grupo. • Deve ser capaz de acessar relatórios das falas dos alunos no fórum e nos Estudos de Caso. • Deve ser capaz de acessar estatísticas sobre freqüência, horários e datas de maior acesso dos alunos, páginas acessadas, configuração do computador usado. • Deve ser capaz gerenciar os assuntos, perguntas e respostas da seção de ajuda do ambiente
Administrador	• Deve ser capaz gerenciar os assuntos, perguntas e respostas da seção de ajuda do ambiente • Deve ser capaz salvar e restaurar dados do banco de dados. • Deve ser capaz de publicar uma tarefa. • Deve ser capaz de gerenciar textos de páginas estáticas. • Deve ser capaz de gerenciar o cronograma dos grupos. • Deve ser capaz
Comum a todos	• Deve ser capaz de utilizar o recurso "onde estou?" para acessar páginas visitadas anteriormente. • Deve ser capaz de visualizar as tarefas vinculadas aos grupos. • Deve ser capaz de visualizar textos em gerado em páginas estáticas. • Deve ser capaz de visualizar os eventos contidos nos cronogramas. • Deve ser capaz de preencher e submeter formulário de contato para os administradores do sistema.

Quadro 3.4 – Novos requisitos funcionais para os diferentes atores do ambiente

3.3.2 – Projeto

De acordo com as abordagens de Pressman (2007) para um projeto de software, decidiu-se que nesta etapa de projeto seriam detalhados apenas os projetos procedimental e de dados. Não houve necessidade de novos projetos de arquitetura e de interface, pois se manteve as especificações definidas na etapa de projeto da versão inicial.

3.3.2.a – Projeto Procedimental

O projeto procedimental desta etapa visa refletir as especificações de requisitos através de digramas UML de casos de uso e atividades. Os diagramas de atividades seguem no anexo eletrônico 1. As descrições dos casos de uso são apresentadas a seguir sequencialmente

acompanhadas de seu respectivo diagrama de casos de uso para os módulos adaptados (principal e estudo de caso) e para os novos módulos especificados (ajuda, backup, tarefas, texto estático, cronograma, contato, personalização da interface, kit pedagógico, usuários online e estatísticas).

Principal: No módulo principal, o [system] do XOOPS determinou-se a inclusão de apenas um caso de uso: "visualizar links para telas visitadas" disponível a todos os usuários (Figura 3.33).
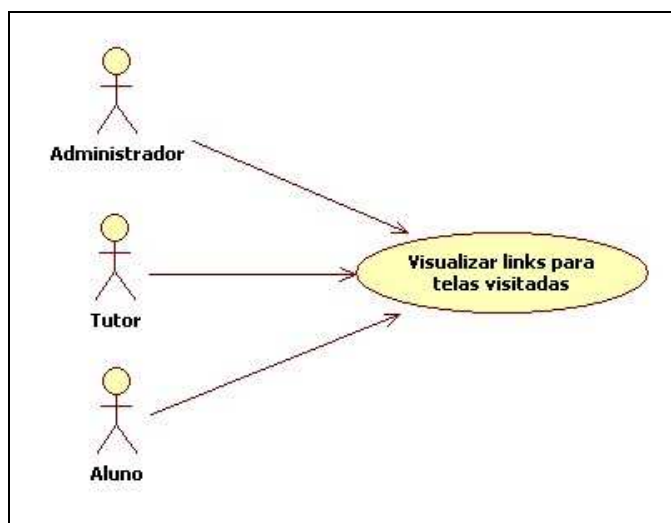
Figura 3.33 – Novo caso de uso para o módulo principal

Estudo de Caso: O módulo de Estudo de Caso é o mais importante dentro do contexto didático-pedagógico do ambiente, portanto foi mais suscetível a mudanças, que os demais. Os novos casos de uso permitem a todos os usuários "visualizar kit pedagógico" e ao professor "fechar grupo de estudo", "gerenciar conteúdo", "gerenciar bibliotecas" e "obter relatórios", conforme apresentado na figura 3.34.
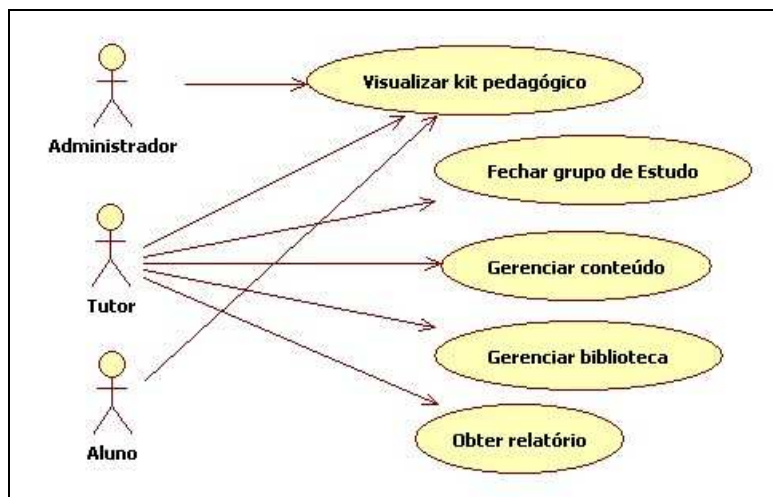
Figura 3.34 – Novos casos de uso para o módulo de Estudo de Caso

Ajuda: As definições do módulo de avisos determinam um módulo em que o professor poderá "gerenciar assunto", "gerenciar pergunta" e "gerenciar resposta" e todos os usuários podem "visualizar assunto", "visualizar pergunta" e "visualizar resposta" (figura 3.35).
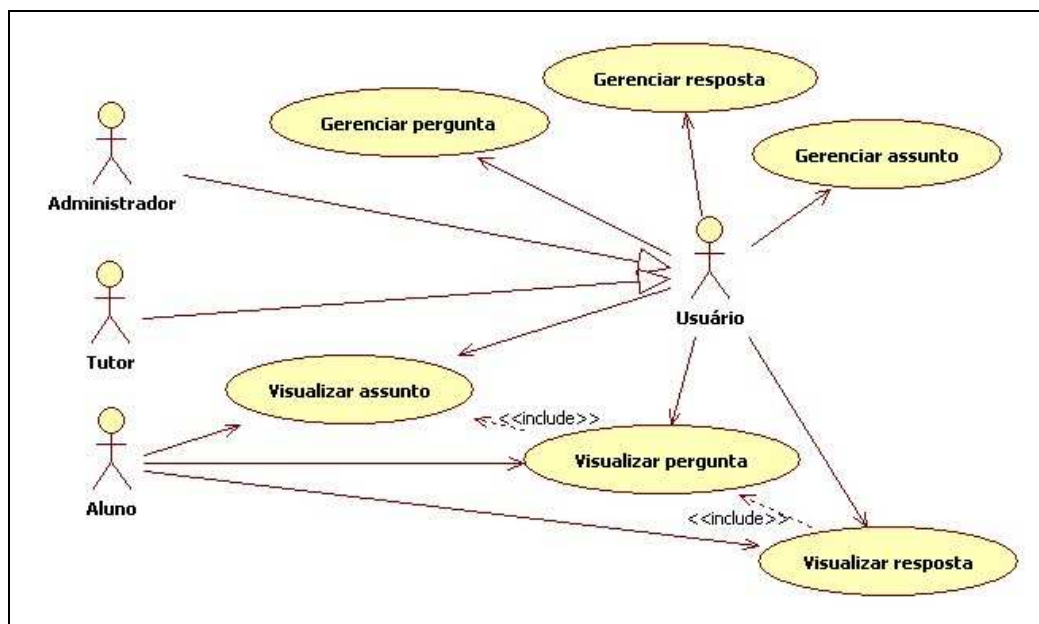
Figura 3.35 – Casos de uso para o módulo de ajuda

Backup: O módulo de backup surgiu da necessidade dos administradores poderem "salvar tabelas do banco de dados" e "restaurar tabelas do banco de dados" (figura 3.36).
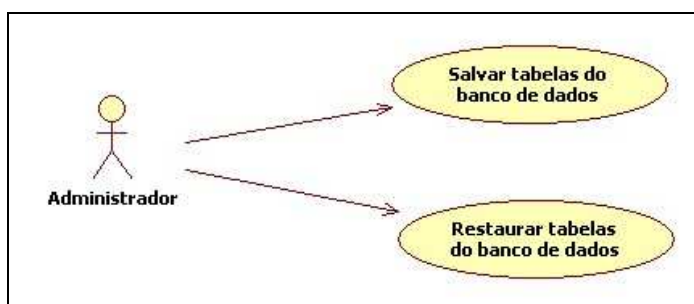
Figura 3.36 – Casos de uso para o módulo de backup

Tarefas: O módulo de tarefas apresenta a possibilidade do professor e administrador "gerenciar as tarefas" e de todos os usuários "visualizar tarefas" conforme o diagrama da figura 3.37.
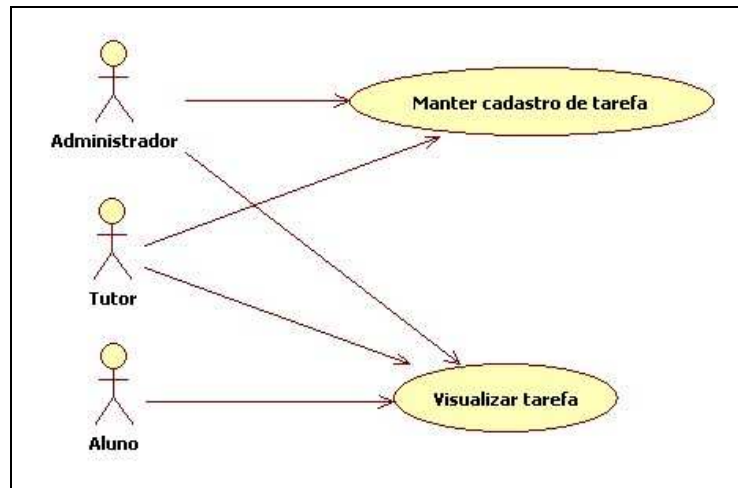
Figura 3.37 – Casos de uso para o módulo de tarefas

Texto estático: O módulo de texto estático surge da necessidade de publicação de textos estáticos sem afetar a estrutura modular do sistema. O módulo permite ao administrador “gerenciar texto” e à todos os usuários “visualizar textos” (figura 3.38).
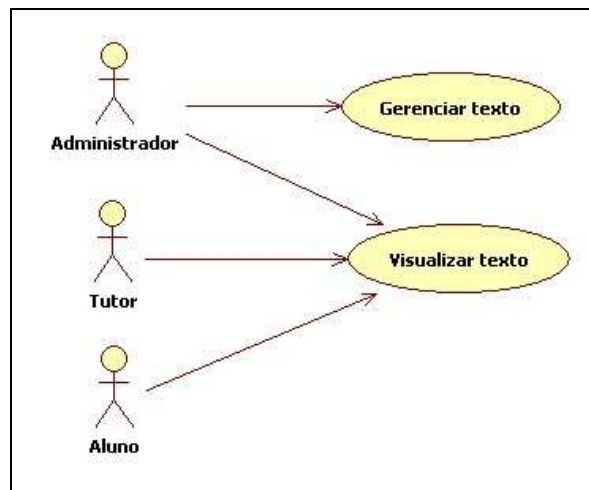
Figura 3.38 – Casos de uso para o módulo de textos estáticos

Cronograma: O módulo de cronograma permite ao professor e administrador “gerenciar eventos”. Os eventos consistem de datas e prazos para execução de tarefas e passos ao longo do estudo, logo todos os usuários podem “visualizar evento” (figura 3.39).
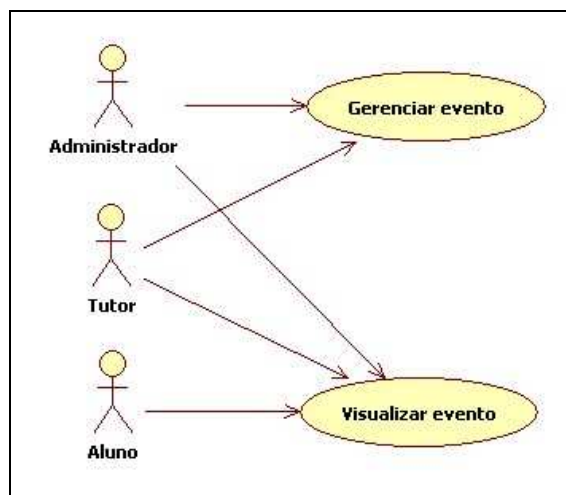
Figura 3.39 – Casos de uso para o módulo de cronograma

Contato: O módulo contato é necessário para que todos os usuários possam “submeter formulário” aos e-mails dos administradores do sistema, contendo alguma dúvida ou sugestão. (figura 3.40).
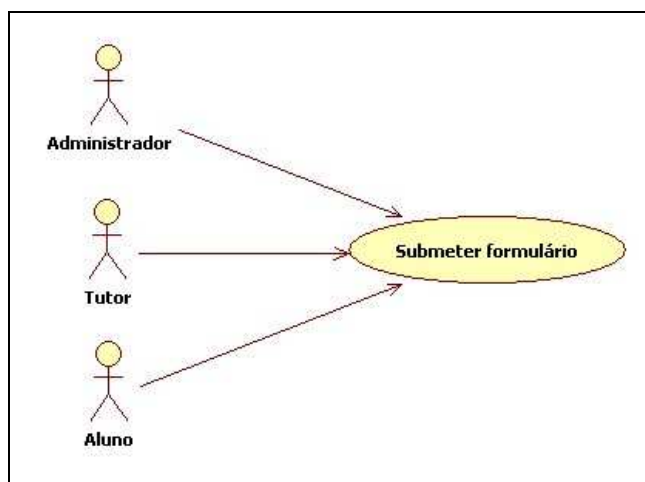
Figura 3.40 – Caso de uso do módulo contato

Personalização da Interface: O módulo destinado a personalização da interface deve permitir que o usuário substitua o tema que compõe sua interface. Para isso é necessário que o desenvolvedor crie várias opções de tema para que através do sistema, o administrador possa “manter temas”. Todos os usuários do sistema terão a possibilidade de “substituir o tema” apresentado.
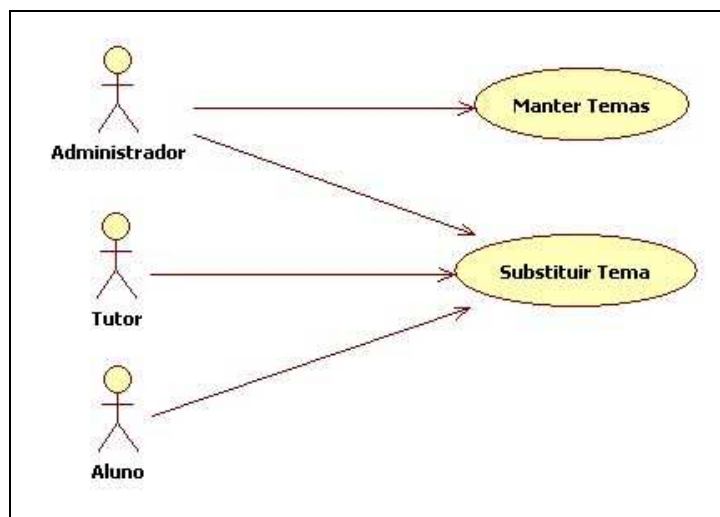
Figura 3.41 – Casos de uso do módulo de personalização de interface

Kit Pedagógico: Os casos de uso para o módulo de kit pedagógico visam viabilizar o cadastro, gerencia e visualização dos materiais didáticos, assim o administrador encarregasse de “gerenciar categorias” de materiais didáticos. Os professores podem “associar material ao grupo”. Professores e administradores podem “gerenciar material no kit pedagógico” e todos os usuários podem “visualizar material” após “visualizar categoria”.
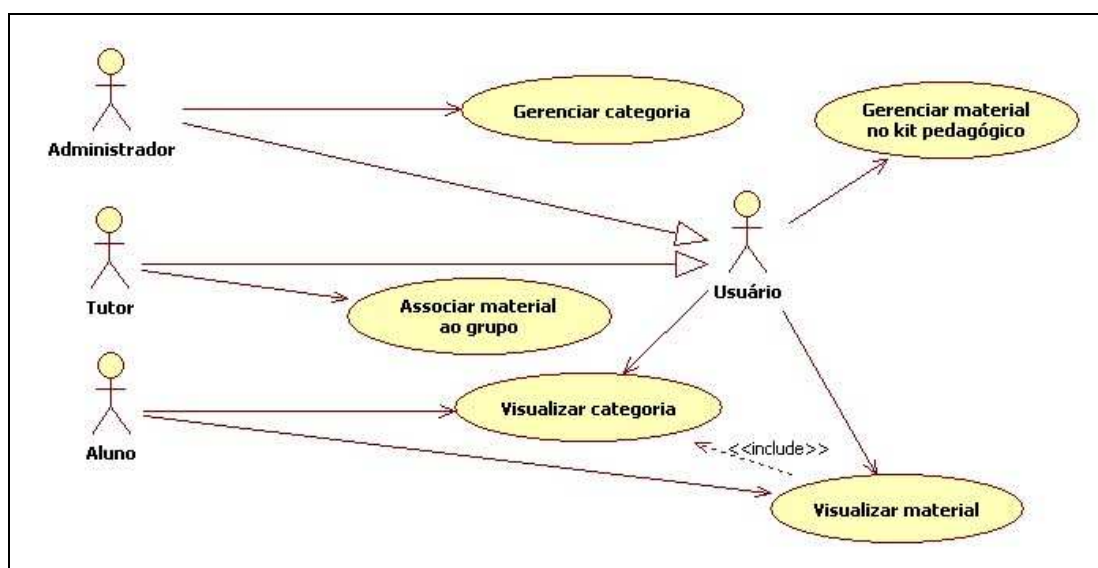
Figura 3.42 – Casos de uso do módulo kit pedagógico

Usuários Online: No módulo usuários online, todos os usuários compartilham um memso caso de uso, que é “visualizar usuário online”.
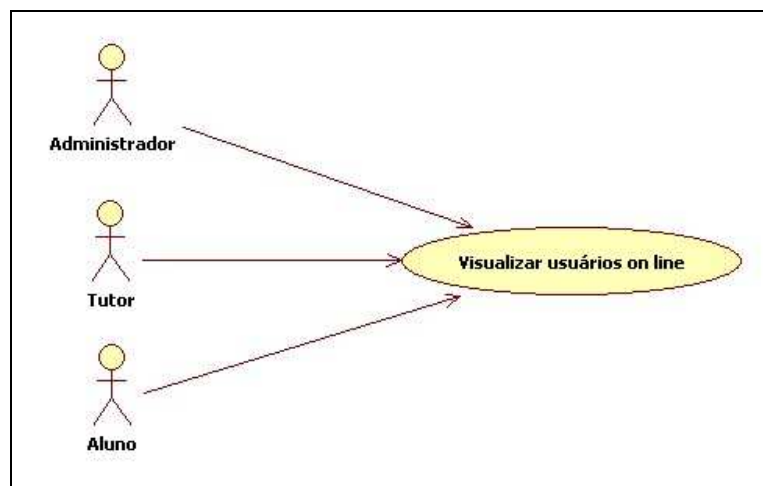
Figura 3.43 – Caso de uso do módulo usuários online

Estatísticas: O módulo de estatísticas deve apresentar diversas estatísticas do sistema, associadas relacionadas individualmente aos usuários ou não. O professor pode, juntamente com o administrador, "exibir estatísticas do aluno", enquanto apenas os administradores podem "exibir estatísticas gerais" do site e "gerenciar as configurações" do módulo.
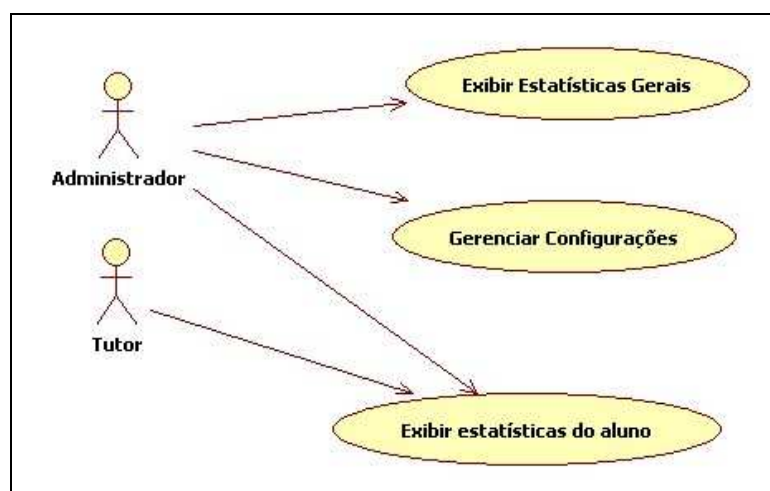
Figura 3.44 – Casos de uso do módulo de estatísticas

3.3.2.b – Projeto de Dados

Segundo Xavier & Portilho (1995), um aspecto fundamental da fase do projeto consiste em estabelecer de que forma serão armazenados os dados do sistema. Para isto, verificou-se durante o desenvolvimento do sistema inicial a adequação de representar o banco de dados através de um DED, já que proporciona a representação das tabelas de dados e seus relacionamentos a partir de modelos conceituais de dados.

O DED do sistema EVA para a codificação das novas funcionalidades modeladas no projeto procedimental realizado na seção 3.3.2.a é apresentado na figura 3.45.
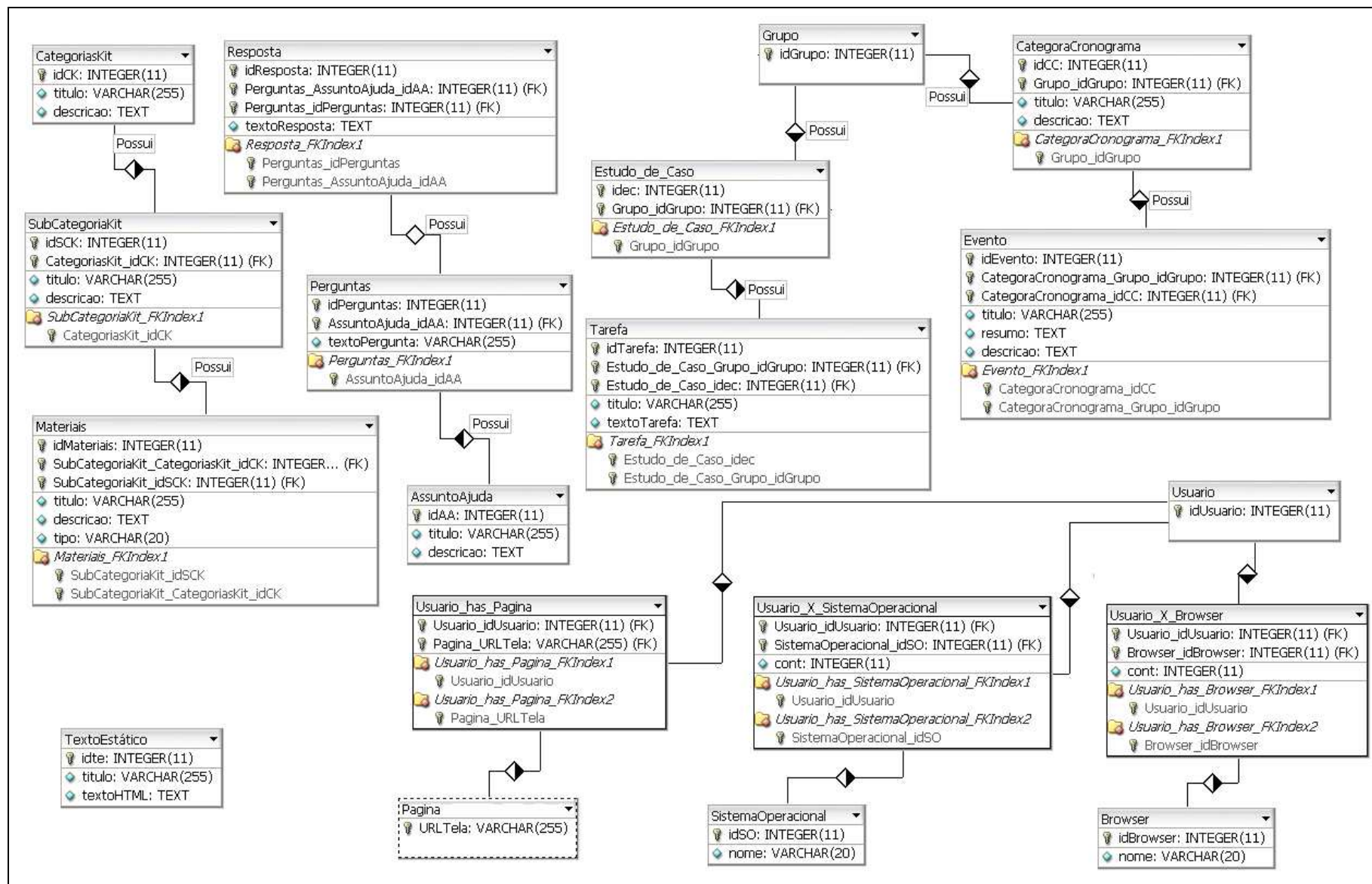
Figura 3.45 – DED dos novos recursos para a versão beta

No gráfico os retângulos representam as tabelas, cada linha do retângulo uma coluna na tabela de dados e as linhas entre as tabelas representam os relacionamentos.

3.3.3 – Codificação

A codificação da versão beta do sistema seguiu duas linhas: a inclusão de novos requisitos nos módulos desenvolvidos e a inclusão de novos módulos.

3.3.3.a – Adaptação dos Módulos Desenvolvidos

Como descrito na remodelagem de escopo, houve a necessidade de adaptar o módulo principal [system] e o módulo de Estudos de Caso [evax_admin]. A seguir é feita a descrição das alterações realizadas.

Principal:

No EVA os usuários podem, a partir da página inicial de um grupo, acessar as ferramentas de fórum, kit pedagógico, avisos, tarefas etc. Após acessar qualquer uma destas áreas, o usuário pode se aprofundar na navegação dentro do módulo.

No EVA o aprofundamento na navegação pode ocasionar ao usuário a não lembrança de todas as páginas visitadas, ou desejar retornar a uma já visitada sem ter de usar o botão “voltar” do navegador. Portanto foi criado um recurso utilizado em portais na Internet denominado “Onde Estou?”. Este recurso organiza no topo da página, em uma linha, o título de todas as páginas visitadas a partir da inicial separadas por uma seta, em ordem hierárquica.

Para desenvolver está ferramenta criou-se no diretório {class/} do XOOPS uma pasta com o nome de {remakepath}, na qual é armazenada uma classe de mesmo nome criada com o objetivo de receber o endereço URL da página e realizar o tratamento necessário para devolver a linha com todos os links relacionados desde a tela inicial até a atual.

A lógica do funcionamento se dá da seguinte forma: a classe {remakepath} recebe a URL que o usuário está acessando e através dela identifica a qual módulo ela pertence (Figura 3.46).

```
<?php
        include ("../../mainfile.php");
        $url = getenv('REQUEST_URI');
        if (strpos($url, "modules/") == true) {
            $module = explode("/", strstr($url, 'modules'));
            if (file_exists(XOOPS_ROOT_PATH."/class/remakepath/paths/".$module[1].".php")) {
                include_once ("paths/".$module[1].".php");
            }
            else {
                include_once ("paths/unknown.php");
            }
        }
        else {
            include_once ("paths/system.php");
        }
        showremakedpath($url);

?>
```

Figura 3.46 – Classe {remakepath}

Esta classe após identificar a que módulo pertence a URL, inclui a classe relacionada ao módulo, que, por fim, devolve todo o caminho de link relacionado àquela página através da função {showremakedpath($url)}.

Na figura 3.47 é exibida uma seqüência de telas do fórum contendo este recurso.
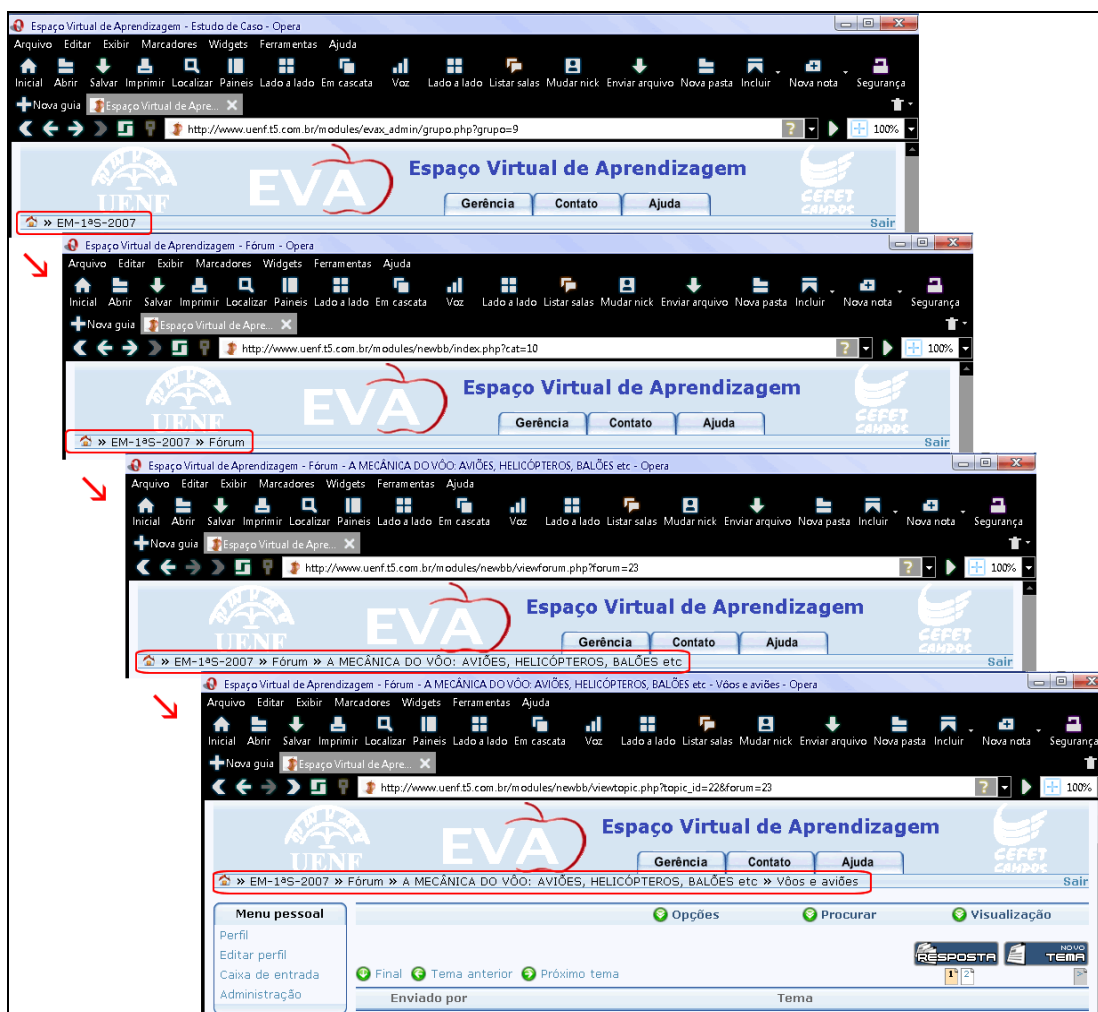
Figura 3.47 – Seqüência de telas do fórum ressaltando o "Onde Estou?"

As quatro partes de tela apresentadas na figura 3.47 se referem a seqüência necessária de acessos para visualização das discussões no fórum. O recurso “onde estou?” está destacado nas telas por um retângulo vermelho, percebe-se que à medida que aumenta se avança na navegação aumenta a quantidade de links. As telas referem-se, respectivamente às telas: inicial do grupo, inicial do fórum, exibição de temas, exibição de discussão.

Estudo de Caso:

A seguir serão apresentados os acréscimos de funcionalidades realizadas no módulo de Estudos de Caso.

Nova tela de acesso às funcionalidades do módulo de Estudos de Caso

Uma falha percebida no módulo de Estudo de Caso era a questão envolvendo a interface das páginas e a navegabilidade entre elas. Neste contexto os maiores problemas existentes era a distribuição dos links administrativos pelas páginas sem padronização e uma tela inicial pouco atrativa e que não diferenciava as permissões de professores e alunos.

A estratégia utilizada para resolver este problema foi criar uma ferramenta que agrupasse todas as a ferramenta de gerência que um tutor tem acesso. Esta ferramenta foi denominada de Ferramenta de Gerencia do Professor.

Outra medida importante neste sentido, foi a reconstrução da tela inicial de um grupo. Esta tela apresenta-se pouco atrativa e não faz clara distinção entre as funcionalidades acessíveis ao professor e ao aluno. Sua reconstrução consistiu na substituição dos links “clique AQUI para...” por ícones acompanhados de uma ou duas palavras representando a função que ele acessa.

Com o surgimento da ferramenta de gerência, houve a necessidade de criar uma tela para o aluno, que possui:

- Nome do grupo, que é um link para uma tela com os detalhes do grupo;
- Lista com todos os Estudos de Caso abertos àquele grupo, sendo cada item da lista um link para exibir o texto do estudo de caso, os passos, as respostas aos passos (somente para o professor), a tela de proposta de solução (somente para o aluno) etc;
- Link para o fórum;
- Link para o chat;
- Link para o cronograma;
- Link para os avisos;

- Link para as tarefas e
- Link para o kit pedagógico.

Esta tela é apresentada na figura 3.48 a seguir.
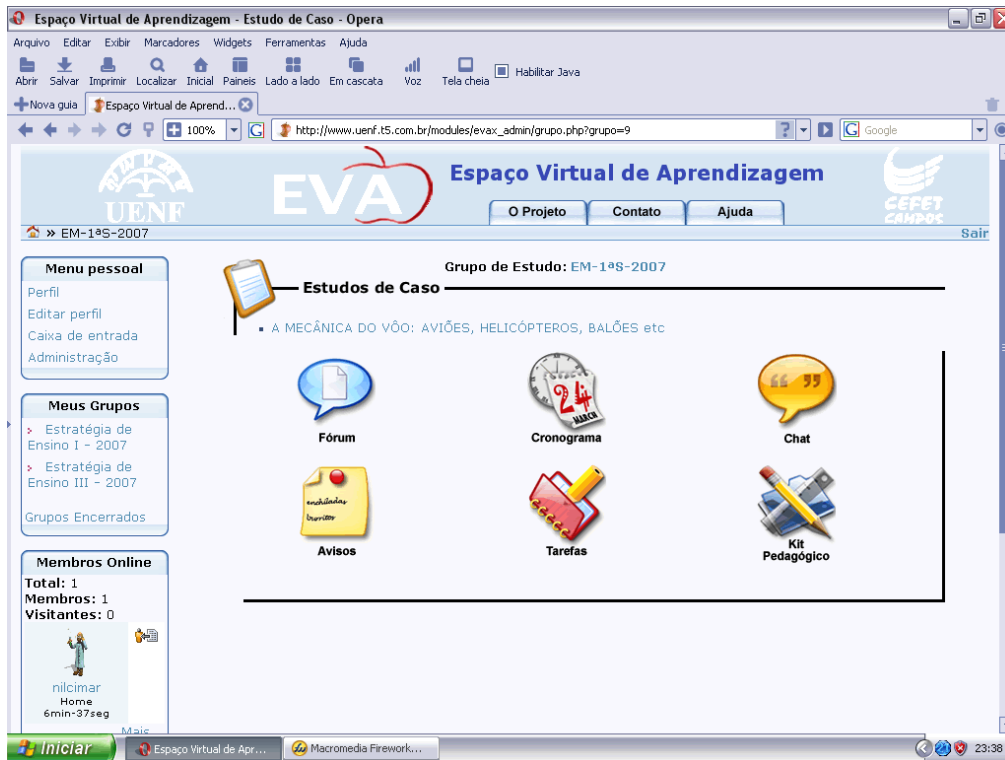
Figura 3.48 – Tela inicial do módulo de Estudos de Caso com a visualização do aluno

O objetivo com os ícones foi facilitar as ligações do estudante com a função de cada ferramenta. Por exemplo, o ícone que apresenta uma régua e um lápis sobre uma pasta, representa o kit pedagógico, já o bloco de anotações representa o módulo de tarefas.

A tela na visão do professor consiste na mesma tela do aluno acrescida de um ícone para a ferramenta de gerência do tutor (figura 3.49).
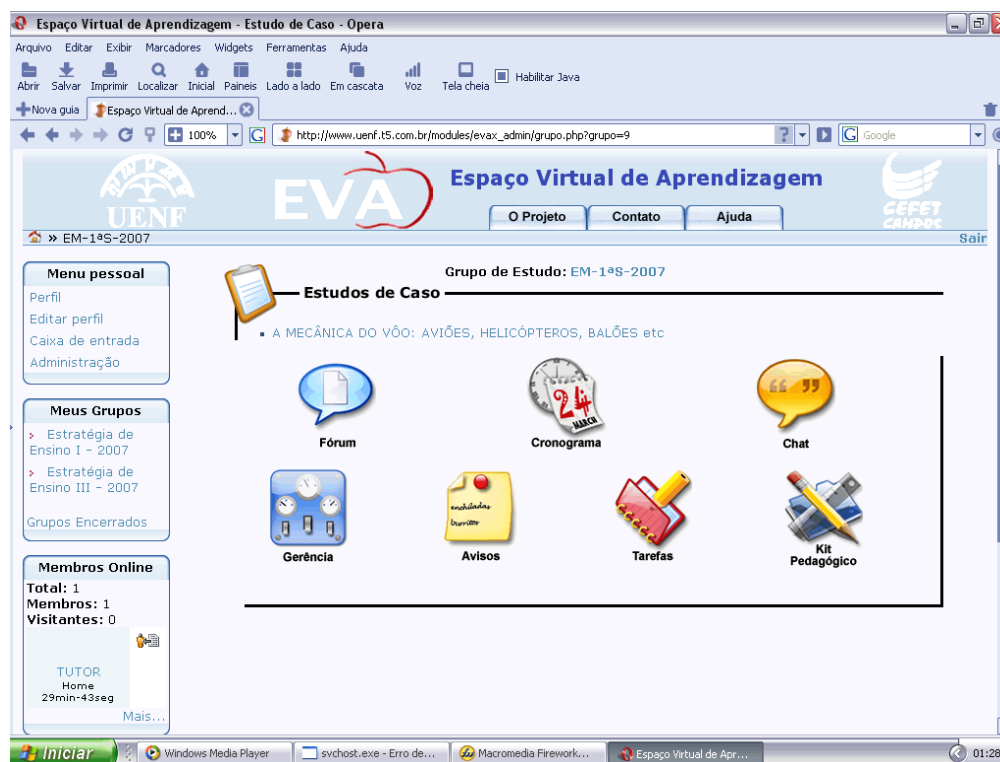
Figura 3.49 – Tela inicial do módulo de Estudos de Caso com a visualização do professor

Destaca-se o ícone gerência, representado por um painel de controle, que permite os professores acessarem a Ferramenta de Gerência do Tutor. Desta forma os professores e alunos possuem, dinamicamente, telas com visões distintas, diferentemente de como era exibido na tela criada para o sistema inicial.

Gerência de grupo

Na escala em que aumentava o número de grupos abertos no EVA, evidenciava-se a necessidade de tornar possível o encerramento das atividades de um grupo, ou seja, fechá-lo de forma que saia do menu “Meus Grupos”. Este menu, situado em todas as telas do EVA, exibi uma lista com os grupos a quais o usuário participa. Um usuário que tenha participado de vários grupos possui em sua tela uma extensa lista com todos eles. Com a opção de fechar um grupo, este deixa de ser exibido no menu “Meus Grupos” e é redirecionado para uma tela específica em que o usuário pode visualizar todos os grupos encerrados no EVA e que ele tenha participado (Figura 3.50).
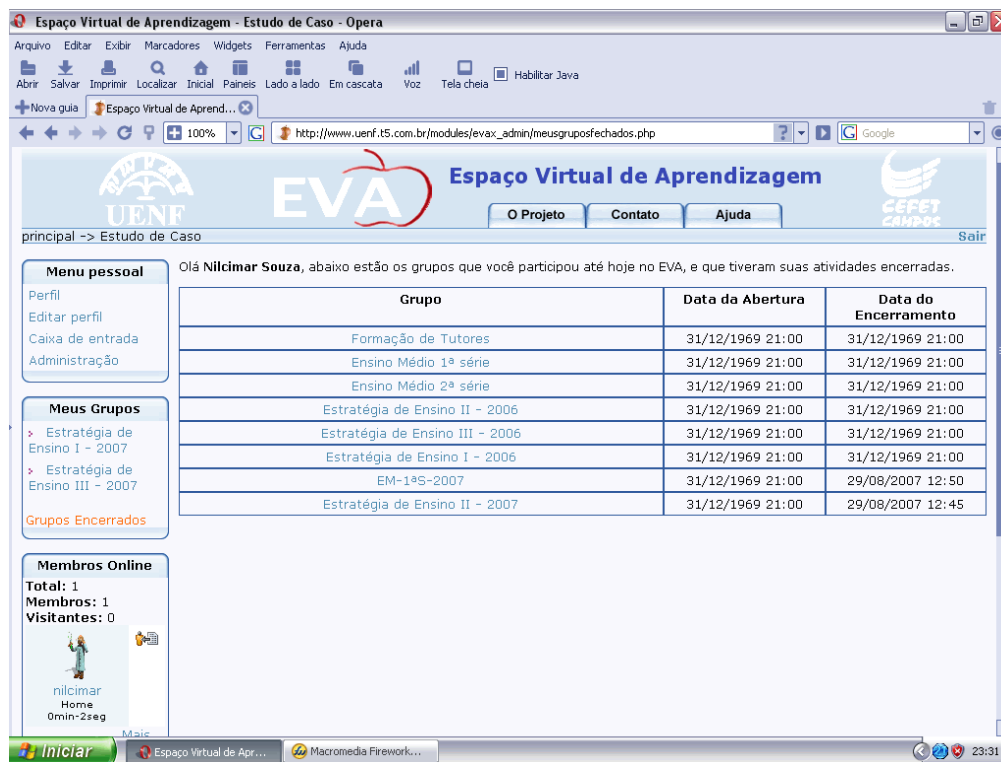
Figura 3.50 – Tela de exibição dos grupos encerrados em que um usuário participou

Definiu-se que um grupo após ser fechado não poderia voltar a ser aberto, esta decisão visou preservar os dados, impedindo-os de sofrer alterações com o objetivo de resguardá-los para fins unicamente de pesquisa. A figura 3.51 exibe a tela em que o administrador pode fechar um grupo.
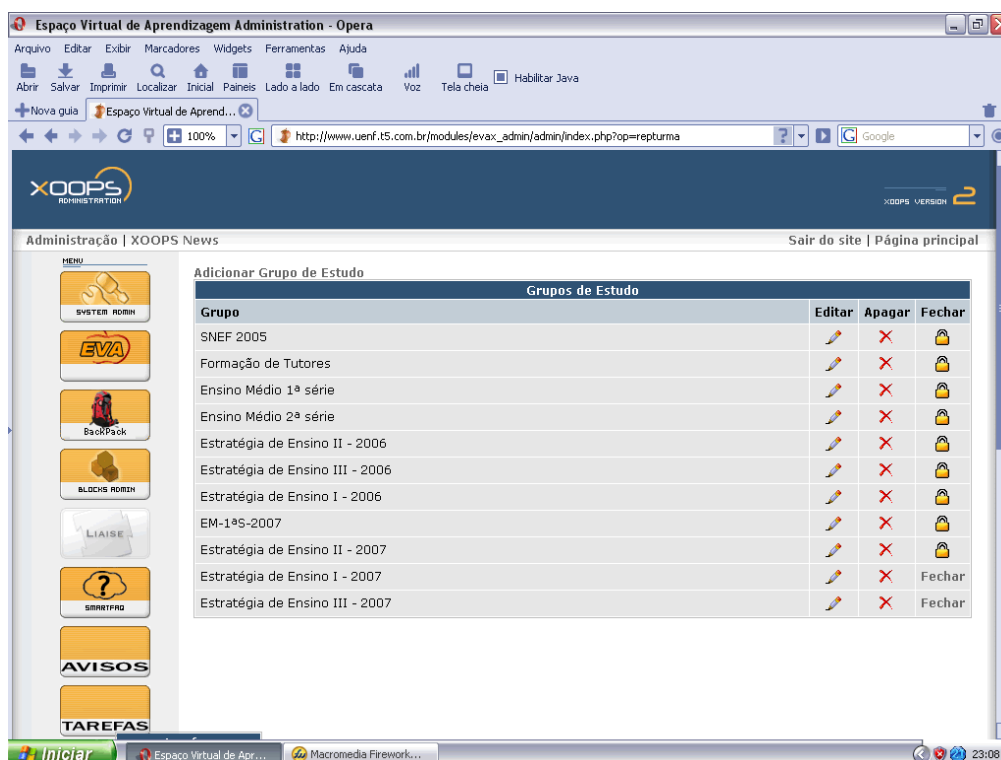
Figura 3.51 – Tela em que o administrador pode fechar um grupo

A tela apresentada na figura 3.51 possui em sua área principal as ações do administrador para gerenciar um grupo: adicionar, editar, apagar e fechar. A opção de adicionar é acessada através do link na parte superior “Adicionar grupo de estudo”. A frente do nome dos grupos criados exibe-se a opção de editar as informações do grupo (ícone de um lápis), apagar o grupo do banco de dados (ícone em forma de “x”) e fechar o grupo, acessado pela palavra “fechar”. Após fechar um grupo, exibe-se o ícone de um cadeado, representando que o grupo está fechado e não é permita sua reabertura.

Ferramenta de Gerência do Tutor

A ferramenta de gerência do tutor surgiu da necessidade de ampliar a independência do professor dentro do grupo, conseqüentemente desvinculando uma grande parte das atividades de gerenciamento do administrador. Assim atividades como incluir, editar e excluir Estudos de Caso e material ao kit pedagógico passam a ser disponíveis, também, aos professores através de uma interface própria e totalmente independente da área administrativa do XOOPS.

Esta ferramenta organiza todas as ações inerentes à administração de um grupo em um único local, dando maior dinamismo e praticidade para o professor no gerenciamento global do grupo. A tela inicial desta ferramenta é apresentada na figura 3.52.
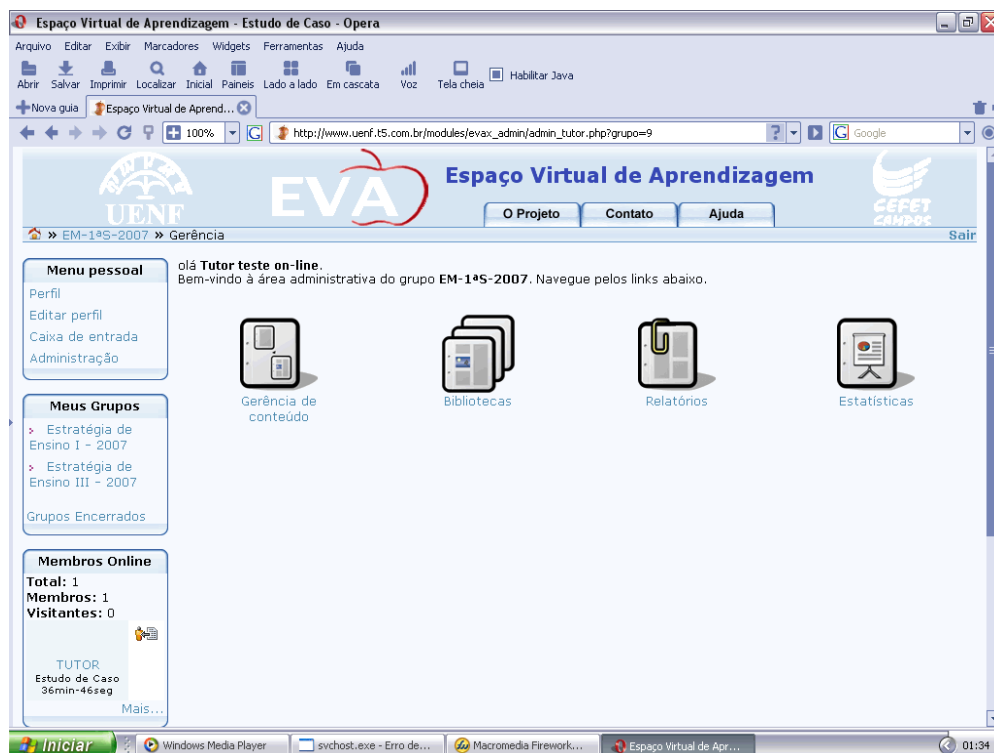
Figura 3.52 – Tela inicial da ferramenta de gerência do tutor

A tela é exibida ao professor após selecionar a opção “gerência” na figura 3.49Ela apresenta as divisões das ações de gerenciamento do professor: gerência de conteúdo (representada pelo ícone contendo duas folhas e uma seta entre eles), bibliotecas (representada pelo ícone de três folhas sobrepostas), estatísticas (representada pelo ícone de um cavalete exibindo um gráfico) e relatórios (representada pelo ícone de uma folha de papel presa por clips). Sendo os 3 primeiros implementados com a instalação de novos módulos. Relatórios é uma ferramenta criada no próprio módulo de Estudos de Caso.

Gerência de conteúdo

No contexto da Ferramenta de Gerência do Tutor a gerência de conteúdo funciona como espaço onde o professor poderá inserir, editar ou excluir um aviso, uma tarefa ou um evento ao cronograma. Além disso, poderá associar, editar e desassociar Estudos de Caso e materiais ao kit pedagógico. As telas específicas para cada uma das atividades de gerenciamento citadas, são acessíveis a partir da tela apresentada na Figura 3.53.
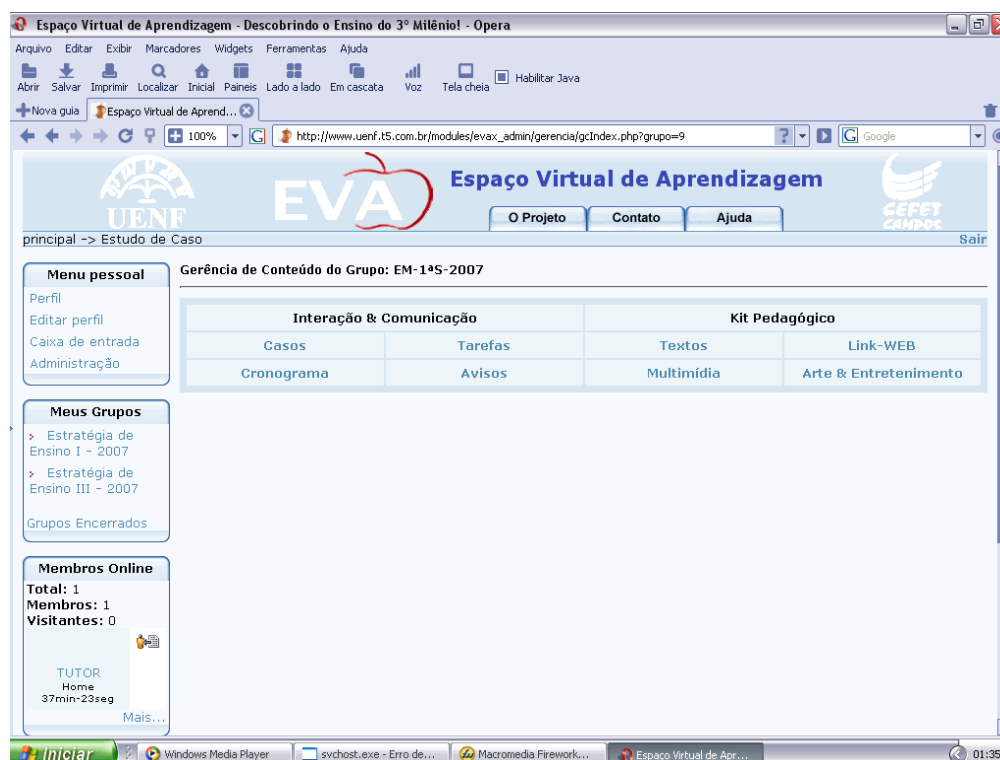
Figura 3.53 – Tela inicial da gerência de conteúdo

Na área principal da tela apresentam-se as opções de gerenciamento da ferramenta. Na coluna à esquerda agrupam-se os módulos referentes à Interação & Comunicação e à direita as opções de gerenciamento do Kit Pedagógico.

O desenvolvimento da gerência de conteúdo se dá a partir de dois módulos, que compõe o kit pedagógico. O processo de adaptação destes módulos é apresentado na seção que aborda a busca, escolha e adaptação de novos módulos.

Bibliotecas

A ferramentas conhecida como bibliotecas (Figura 3.54) foi projetada para ser auxiliar à ferramenta de gerência de conteúdo. Como mostrado anteriormente, a gerência de conteúdo permite a associação de Estudos de Caso e materiais ao kit pedagógico. A inclusão direta do material deve ser feita dentro do espaço bibliotecas, onde ficarão armazenados todos os estudos de caso e materiais do kit pedagógico para que, através do espaço gerência de conteúdo, sejam associados aos grupos. Desta forma, um mesmo estudo de caso ou material para o kit pedagógico não necessita ser incluído no banco de dados todas as vezes que for utilizado. O professor utiliza este espaço para cadastrar o material ou o estudo de caso, e em seguida, sempre que precisar os utiliza em algum grupo bastando acessar a área de gerência de conteúdo e associá-lo ao grupo. Ou o material ou estudo de caso é armazenado uma única vez, mesmo sendo usado por grupos e professores diferentes.
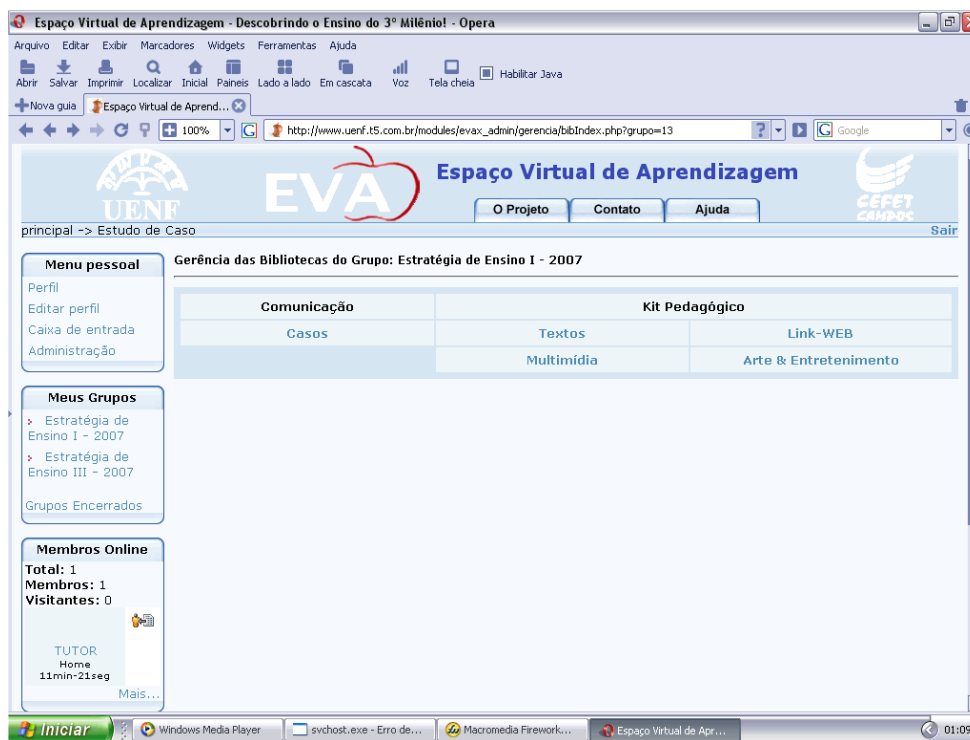
Figura 3.54 – Tela inicial das bibliotecas

Na área principal da tela apresentam-se as opções de gerenciamento das bibliotecas. Na coluna à esquerda o gerenciamento da biblioteca de Estudos de Caso. Na coluna à direita as bibliotecas do kit pedagógico.

O desenvolvimento da ferramenta bibliotecas se deu a partir dos módulos que compõe o kit pedagógico. Isto representa as relações existentes entre as ferramentas de bibliotecas e de gerência de conteúdo.

Estatísticas

A ferramenta de estatística permite que o professor obtenha informações como freqüência, horários e datas de maior acesso, páginas acessadas, configuração do computador usado pelo usuário etc. Estas informações poderão ser visualizadas por aluno ou geral. No contexto da ferramenta de gerência do tutor, as estatísticas são extraídas a partir da adaptação de um módulo. O processo de adaptação deste módulo é apresentado na seção que aborda a busca, escolha e adaptação de novos módulos.

Relatórios

A ferramenta conhecida como relatórios permite a extração de relatórios das falas dos estudantes no meio de comunicação fórum e interação Estudos de Caso. A extração das falas é fundamental para a análise e avaliação das mesmas. Os relatórios passíveis de extração são os relatórios de do fórum, categorizados por aluno ou por tópico e, as das falas dos alunos nas respostas aos passos dos casos separados por aluno ou por passo (Figura 3.55).
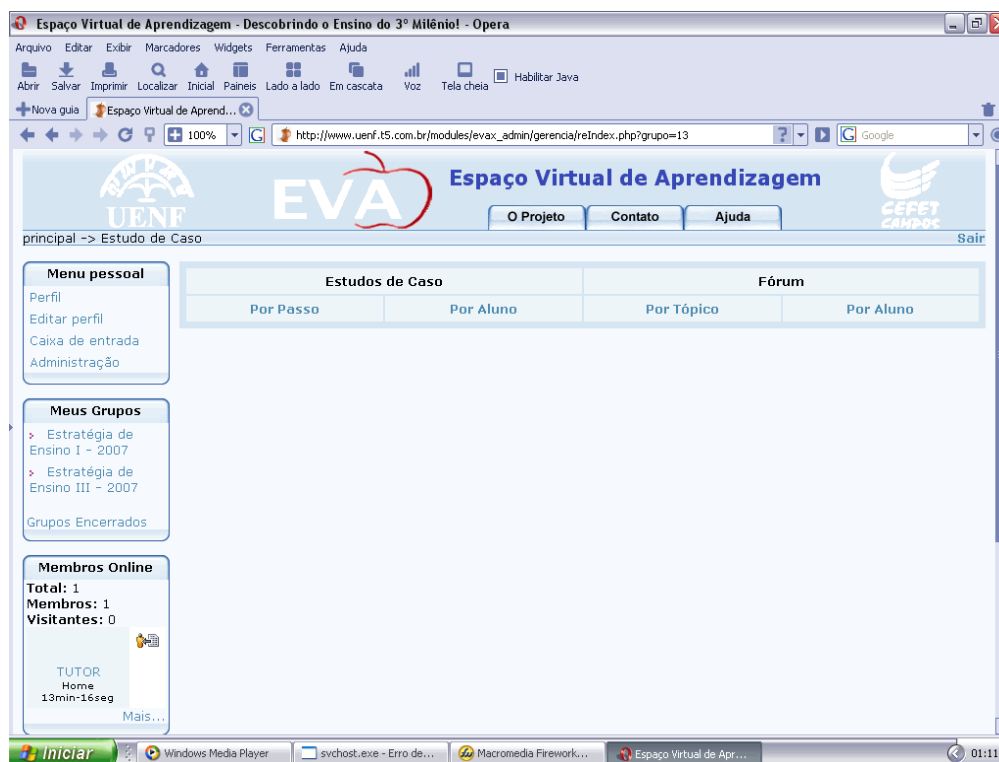
Figura 3.55 – Tela inicial dos relatórios

Na área principal da tela apresentam-se as opções de obtenção de relatórios. Na coluna à esquerda os relatórios do módulo de Estudos de Caso e na coluna à direita os relatórios do módulo de fórum.

O desenvolvimento desta ferramenta não foi realizado a partir da adaptação de nenhum módulo pré-existente, ele foi codificado integralmente como uma funcionalidade do módulo de Estudo de Casos e apresentado em Sepulvida (2008).

3.3.3.b – Busca, Escolha e Adaptação de Novos Módulos

O paradigma híbrido adotado para construção do sistema beta teste, o modelo evolucionário exploratório baseado em reuso de componentes, sugere a adoção do reuso de componentes na etapa de codificação. A metodologia adotada no desenvolvimento do sistema inicial foi utilizada para buscar, escolher e adaptar os novos módulos da versão beta. Houve u um levantamento por módulos que implementassem as funcionalidades modeladas na etapa de projeto ou que mais se aproximassem. O levantamento aconteceu nas principais comunidades XOOPS na internet.

Os módulos alvo da pesquisa relacionaram-se com os projetados, assim o levantamento consistiu em buscar módulos com características de ajuda, backup, tarefas, texto estático, cronograma, contato, personalização da interface, kit pedagógico, usuários online e estatísticas. A seguir é apresentada uma breve descrição de cada um dos módulos selecionados durante o levantamento e qual a funcionalidade que ele implementa dentro da estrutura. São apresentadas, também brevemente, as modificações realizadas no módulo para se encaixar na especificidade do sistema.

Backup

O servidor que hospedava o EVA não está acessível fisicamente aos administradores, desta forma a única via de acesso é através da interface administrativa disponibilizada pela empresa de hospedagem. Aí surgiu a necessidade, durante a pré-testagem, dos desenvolvedores realizarem backups dos dados no banco sem estar vinculada a interface administrativa do servidor que hospeda o EVA. O objetivo era que os administradores tivessem acesso aos dados no banco para backup e restore sem a necessidade de usar ferramentas como o phpmyadmin.

A solução desta necessidade deu-se através da instalação de dois módulos. Um denominado [backpack] da comunidade japonesa e desenvolvido pelo grupo bluemoon incorporation. Outro denominado [backup] do desenvolvedor suíço Peter.

Utilizou-se os dois módulos, pois ambos não implementavam, sozinhos, todos os recursos necessários. O módulo japonês permite a restauração e backup de dados em uma parcela de tabelas selecionadas pelo administrador ou de todas as tabelas, no entanto o módulo não suporta grande volume de dados, tornando-se adequado apenas para backup e restauração pontuais, mas não te todo o banco de dados (figura 3.56). Neste sentido surge o módulo suíço para realização de backup completo dos dados. Os dois módulos permitem a configuração de qual a forma de compactação dos dados deseja-se obter após o backup: gzip, zip ou sql.
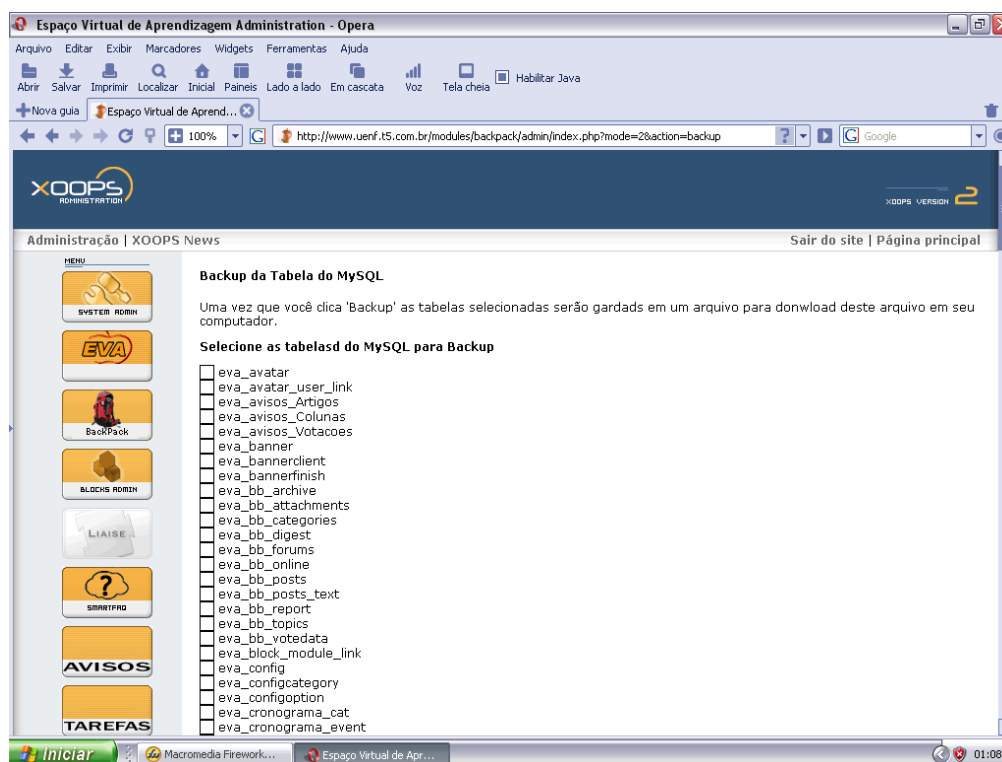
Figura 3.56 – Tela de realização de backups e restaurações pontuais

A tela apresentada disponibiliza em forma de lista todas as tabelas do banco de dados do sistema. O administrador marca as tabelas necessárias para o backup, seleciona o formato de compactação e os salva.

Tarefas

A procura realizada com o objetivo de encontrar módulos com características de um módulo de tarefas revelou a inexistência de um módulo com essa especificidade nas comunidades XOOPS consultadas.

Um novo levantamento foi realizado com o objetivo de localizar um módulo com características aproximadas de um módulo de tarefas, que fosse passível de adaptação. O

módulo que se apresentou mais próximo da configuração pretendida foi o módulo utilizado para implementação do módulo de aviso, o módulo nacional de notícias [eNoticias]. A estrutura deste módulo se mostrou adaptável à um módulo de aviso e nesta etapa foi utilizado para adaptar-se a um módulo de tarefas. As estruturas são correlatas, o módulo de aviso permite ao professor e administrador gerenciarem os avisos e a todos os usuários visualizar-lhes, esta é a estrutura do módulo de tarefas, altera-se apenas o cadastro e visualização de avisos por tarefas.

As três tabelas de banco de dados do módulo original [eNoticias] ({enoticias_artigos}, {enoticias_colunas} e {enoticias_votacoes}) representam uma coleção de notícias relacionada a uma categoria principal (colunas). A adaptação deste módulo à um módulo de tarefas seguiu os procedimentos utilizados na adaptação para um módulo de avisos. Neste caso, há uma coleção de tarefas relacionadas a um grupo de estudo.

A mudança conceitual do módulo foi realizada através de renomeação de tabelas no banco de dados. Uma vez que esse módulo segue o mesmo módulo utilizado para o desenvolvimento do módulo de avisos, é necessária a renomeação das tabelas e pastas para evitar conflitos de reconhecimento de dados. Outra adaptação foi a reconstrução de toda a interface gráfica visando a apresentação do conteúdo de maneira adequada as definições do módulo.

Uma adaptação importante no módulo foi no tocante a permissão. O módulo permitia que se atribua permissões de acesso as categorias de acordo com os grupos existentes. Desta forma solucionou-se a questão de permissões da seguinte forma: ao se criar um grupo, cria-se automaticamente uma categoria no módulo com exatamente o mesmo nome do grupo. Assim é possível o administrador aplicar a permissão para um grupo acessar apenas as tarefas contidas na categoria relacionada ao seu grupo de estudo.

Texto Estático

Durante o pré-teste percebeu-se a necessidade de criar telas com informações, mas que não estivessem ligadas ao conteúdo de nenhum módulo. Porém não se podia abandonar a estrutura modular para que fosse possível a criação de telas estáticas escritas em HTML. Iniciou-se busca por módulo que permitisse o armazenamento de textos em HTML para exibição em telas estáticas.

Verificou-se a existência de vários módulos destinados a implementação desta função, no entanto o selecionado foi o nacional [xt_conteudo] desenvolvido pela empresa Mastop

InfoDigital. Ele permite que o administrador armazene códigos telas inteiras escritas em HTML e aplique as permissões de acesso à tela criada.

Com ele criou-se as telas “Quem somos”, que apresenta os pesquisadores envolvidos no projeto, “normas de conduta”, que apresenta as normas de conduta para a utilização do ambiente, tecnologia, que descreve a tecnologia utilizada no desenvolvimento do EVA e metodologia, que expõe a metodologia utilizada na aplicação dos Estudos de Caso.

Neste módulo não se realizou alterações estruturais, visto que ele apenas exibe o conteúdo criado, ele não possui a responsabilidade de formatá-lo. Houve a remoção de algumas funcionalidades como avaliação e comentários do conteúdo pelos usuários. A figura 3.57 apresenta uma tela gerada pelo módulo.
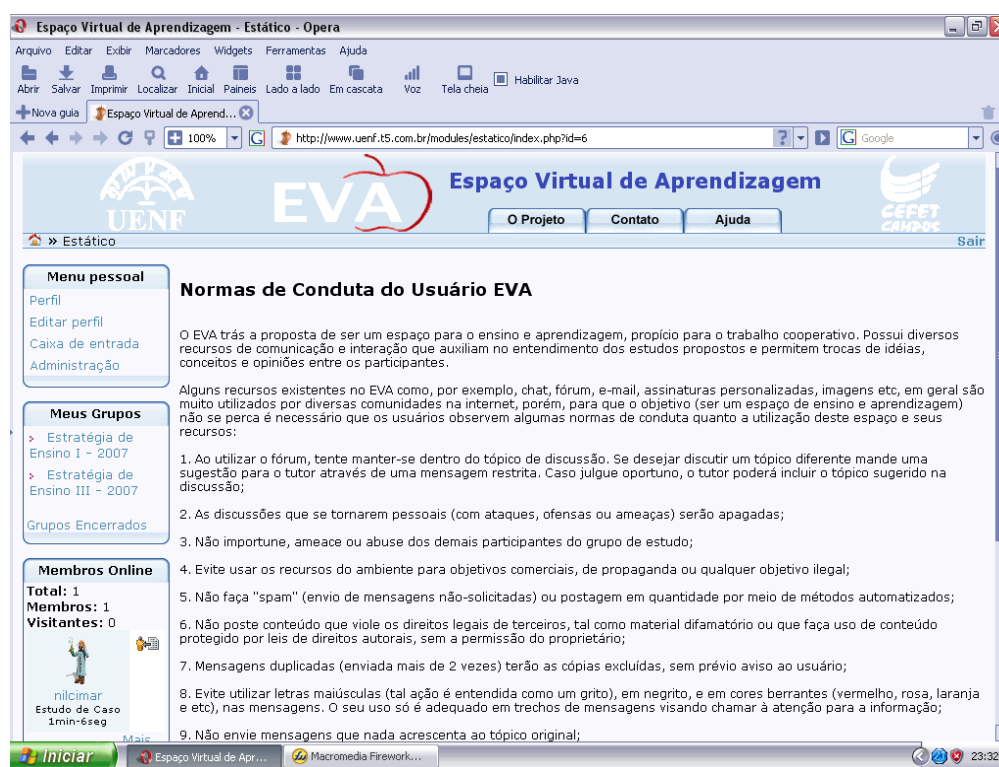
Figura 3.57 – Tela das normas de conduta do usuário gerada pelo módulo de texto estático

Devido à natureza do conteúdo contido nestas telas, as permissões de acesso foram de que qualquer usuário pode acessar todas as telas criadas no módulo.

Cronograma

No pré-teste verificou-se a necessidade de uma ferramenta que auxiliasse na organização do grupo, estabelecendo datas e prazos para execução das tarefas e realização dos eventos. Uma ferramenta com tais características é um cronograma.

Após determinar a necessidade de implementar um módulo de cronograma, realizou-se um levantamento por módulos que adequassem a estas necessidades. O módulo que mais se adequou às necessidades foi o [eXtCal], desenvolvido pelo francês conhecido como Zoullou.

Este módulo executa as ações básicas esperadas para o módulo, ou seja, permitir o cadastro de eventos aos professores e acesso aos eventos pelos integrantes do grupo. Além dessas operações, o módulo permite relacionar os eventos cadastrados à uma categoria e atribuir permissões de acesso a ela (figura 3.58).
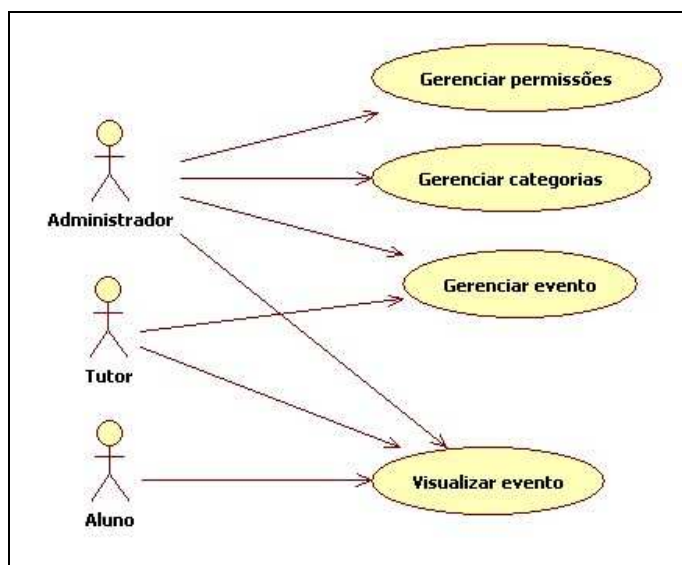
Figura 3.58 – Casos de uso do módulo [eXtCal]

O diagrama da figura 3.58 comparado ao diagrama projetado para este módulo na figura 3.38, revela a total compatibilidade deste módulo com o projeto estabelecido inicialmente. O módulo [eXtCal] permite a atribuição de permissões de acesso as categorias de eventos criadas, assim como medida de restringir cada evento criado a um grupo específico, iniciou-se as adaptações envolvendo as permissões: ao se criar um grupo, cria-se automaticamente uma categoria no módulo com exatamente o mesmo nome do grupo. Assim é possível o administrador aplicar a permissão para um grupo acessar apenas os eventos contidos na categoria relacionada ao seu grupo de estudo.

A figura 3.59 apresenta a tela inicial do módulo de cronograma.
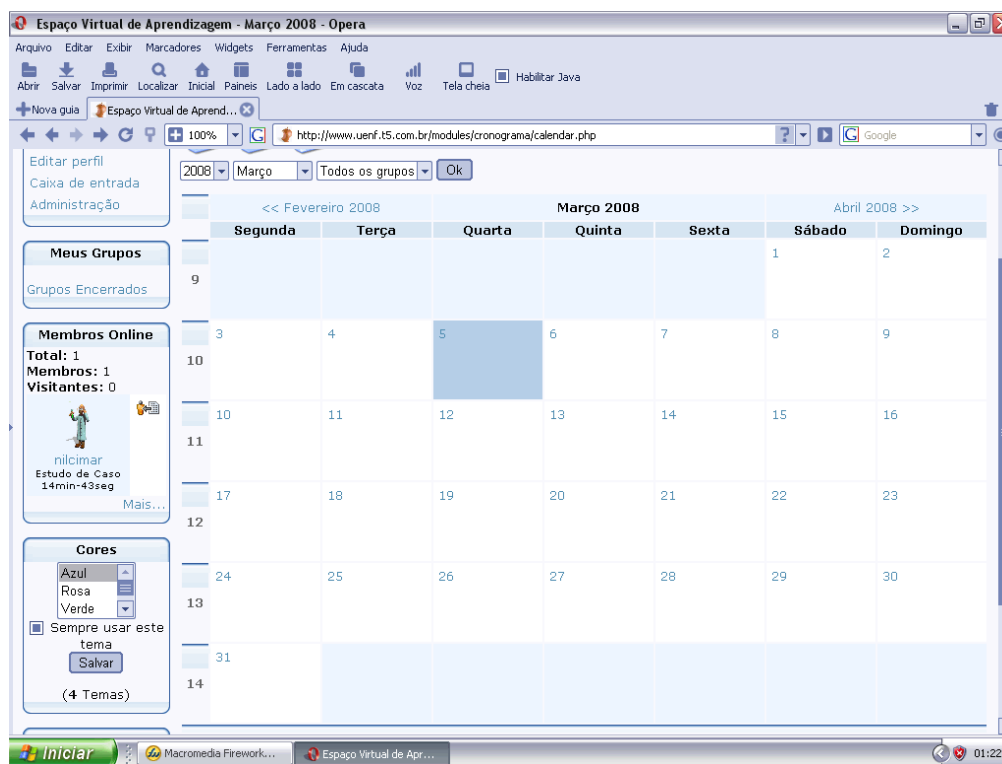
Figura 3.59 – Tela inicial do módulo de cronograma

A tela apresenta um cronograma no formato de calendário. Por padrão é exibido por mês, porém pode ser exibido por ano ou dia. No formato mensal há um destaque para o dia em que o calendário está sendo visualizado e em cada dia podem ser relacionados eventos.

Ajuda

À medida que aumenta o número de usuários no EVA, torna-se estratégico o recurso de ajuda para os usuários, não somente uma ajuda navegacional, como também uma ajuda pedagógica aos professores.

Com o surgimento da necessidade de implantar um módulo de ajuda, realizou-se busca para encontrar algum módulo de ajuda já existente. Dentre os módulos encontrados o escolhido foi o canadense [smartfaq], desenvolvido pela equipe The SmartFactory. Apesar deste módulo ser originalmente um módulo de FAQ, mostrou-se o mais apropriado pela sua estrutura e recursos.

A maior mudança feita neste módulo foi a reconstrução de toda a interface. Na original os acessos do usuário eram compostos de duas telas, sendo que a inicial continha os assuntos, perguntas e respostas e ao clicar em um assunto o usuário era direcionado a uma outra tela com a mesma estrutura da inicial, com uma única diferença, que era exibir as perguntas e respostas do assunto escolhido. Na modificação manteve-se a tela inicial exibindo todos os assuntos, que no EVA se apresentam divisíveis em dois: navegação e didática. Retirou-se as

perguntas e respostas, informações desnecessárias e alterou-se o layout das telas. A mesma modificação foi realizada na tela que exibe as respostas, que passou a exibir apenas as perguntas e o assunto a que ela pertence e não mais as respostas. Para exibir as respostas das perguntas selecionadas pelos usuários foi criada mais uma tela para exibir as respostas das perguntas selecionadas pelos usuários.

Uma das telas modificadas, a tela de assuntos, é apresentada com sua aparência antiga e atual na figura 3.60.
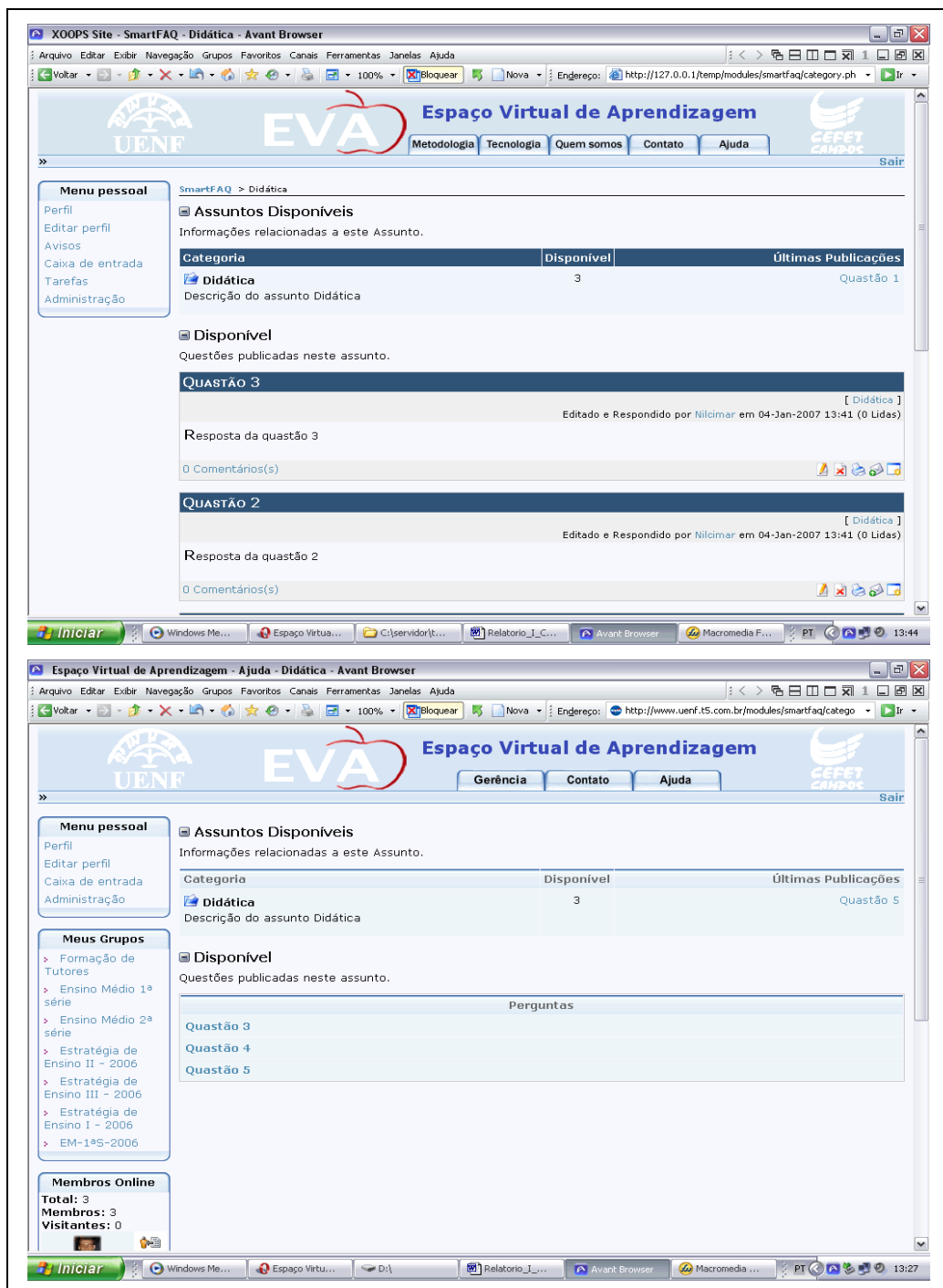
Figura 3.60 – Antiga e atual tela de assuntos do módulo de ajuda

A tela apresentada na parte superior da figura exibe os assuntos, questões e respostas na interface original do módulo [SmartFAQ]. Na segunda tela apresentada na figura são exibidas apenas as questões relativas a um assunto através da nova interface criada. Assim, removeu-se as respostas e assuntos desta tela, de forma a organizar a seqüência de telas e conteúdo no módulo. O módulo permite que se atribua permissões de acesso aos assuntos, de forma que cada grupo possua visões diferentes da ajuda. Desta forma é possível deixar a ajuda do professor visível apenas aos professores e a dos alunos visível a estes.

Contato

O módulo de contato surge da necessidade de os administradores criarem formulários mais facilmente, mais especificamente, um formulário de contato, no qual qualquer usuário pode enviar preencher os dados e submetê-lo. Um formulário com essas características pode ser construído facilmente utilizando uma linguagem de programação Web, mas optou-se novamente por preservar a estrutura modular do sistema, portanto buscou-se módulos com estas características.

Verificou-se que desde as primeiras versões do XOOPS já havia módulos com destinado a esta funcionalidade, no entanto o selecionado foi um módulo mais atual desenvolvido pela empresa Brandycoke Productions denominado [liaise]. Este módulo permite que sejam criados formulários através de uma tela em que os administradores selecionam os componentes e os nomeiam.

As opções de configurações do módulo são: atribuir permissões dos grupos que poderão acessar o formulário, configurar quem receberá os dados submetidos através do formulário e onde serão recebidos, no e-mail ou na caixa de entrada.

Para as permissões de acesso decidiu-se por disponibilizar o módulo a todos os usuários, já que ele é utilizado como meio de submissão de dúvidas, sugestões, comentários etc. Para isso é necessário escrever, juntamente com a mensagem, o nome, e-mail e assunto (figura 3.61). Nas configurações relativas ao recebimento das mensagens, decidiu-se que os dados preenchidos serão recebidos por todos os administradores em seus e-mails.

Não se fez adaptações significativas neste módulo quanto a estrutura e aparência, apenas a adequação de alguns termos traduzidos do inglês.
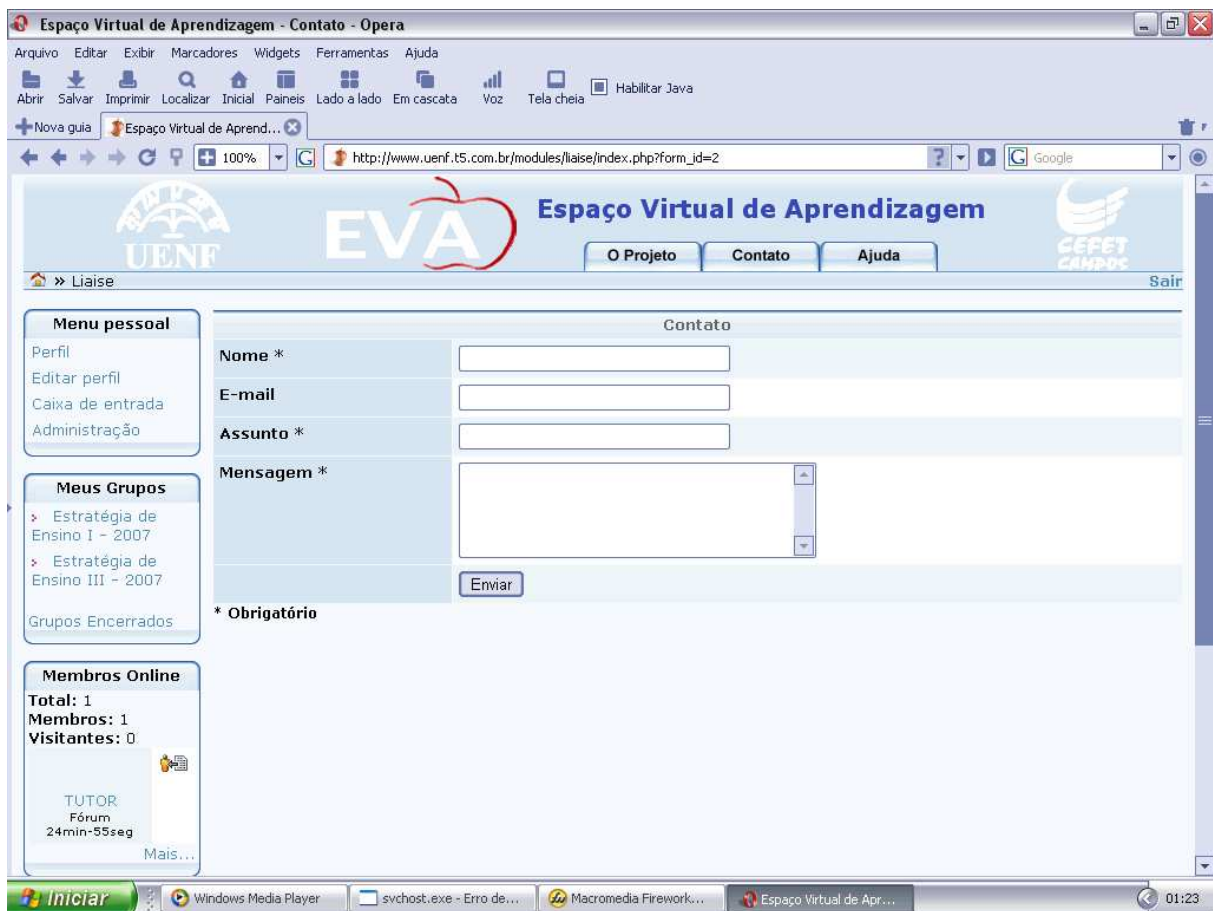
Figura 3.61 – Tela com formulário de contato dos usuários com os administradores

Os campos “Nome”, “Assunto” e “Mensagem” devem ser obrigatoriamente preenchidos antes do envio e opcionalmente pode ser preenchido o campo E-mail.

Personalização da Interface

Com o objetivo de deixar a interface mais adaptável e agradável aos usuários, optou-se por desenvolver temas de cores diferentes, que pudessem ser escolhidos pelos usuários para a apresentação das informações. Foram desenvolvidos quatro temas de cores diferentes, embora todos os temas possuam a mesma estrutura. O design da interface do EVA “*foi elaborado com o objetivo de organizar as informações de forma que a navegação ficasse intuitiva e desse uma noção implícita dos diferentes grupos de ferramentas*” (Araújo, 2005), ressaltando o papel do usuário dentro do ambiente e seu conjunto de permissões.

Para a codificação dos temas tomou-se como base o tema pré-definido default, que utiliza a estrutura de arquivos individualizados, que nos mostrou ser a mais adequada, já que proporciona facilidade de codificação e manutenção. Esta estrutura é representada por um diagrama de componentes na figura 3.62, onde se percebe a presença do arquivo central {theme.html} e sua dependência por 5 outros arquivos no formato HTML:

{theme_blockleft.html}, {theme_blockcenter_l.html}, {theme_blockcenter_c.html}, {theme_blockcenter_r.html}, {theme_blockright.html}, que representam as 5 colunas compreendidas entre o cabeçalho e o rodapé.

O theme.html possui a necessidade de um arquivo {style.css}, que contém as classes CSS responsáveis por controlar fontes, cores, margens, linhas, alturas, larguras, posicionamentos, imagens de fundo etc. da interface. No tema desenvolvido para o EVA sob a estrutura do tema default XOOPS há a necessidade de um pacote de imagens utilizadas para compor a interface, estas imagens localizam-se em um diretório denominado {images/} dentro da pasta do tema e é acessada por todos os arquivos HTML e pelo {style.css}.
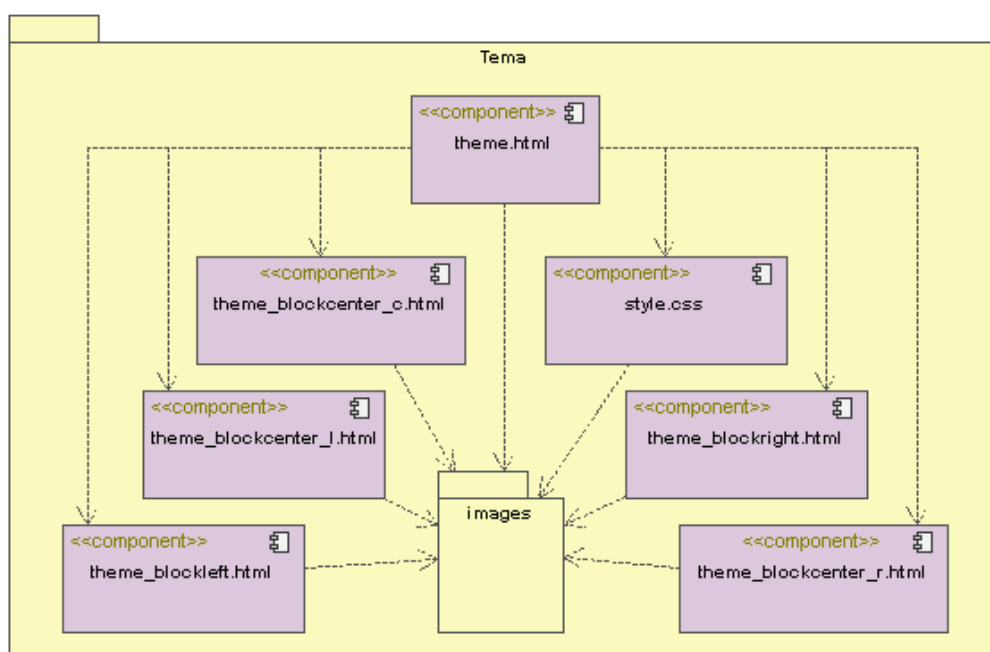
Figura 3.62 – Diagrama de componentes dos temas do EVA

As cores predominantes escolhidas para o tema principal do EVA foram o azul e o branco. Os demais temas criados seguem as seguintes combinações de cores: verde e branco, rosa e branco e vermelho e branco, porém o tema azul é o tema padrão do EVA.

Para que os quatro temas criados fossem disponibilizados de forma que o usuário pudesse escolher um dentre os quatro, foi realizada procura por um módulo que implementasse está funcionalidade. O único módulo com este objetivo é o módulo desenvolvido pelo brasileiro Fernando Santos, [xt-temas], que permite a organização dos temas de forma que o usuário possa escolher qual deseja utilizar em sua interface. A figura 3.63 mostra uma mesma tela sendo exibida nas quatro combinações de cores disponíveis.
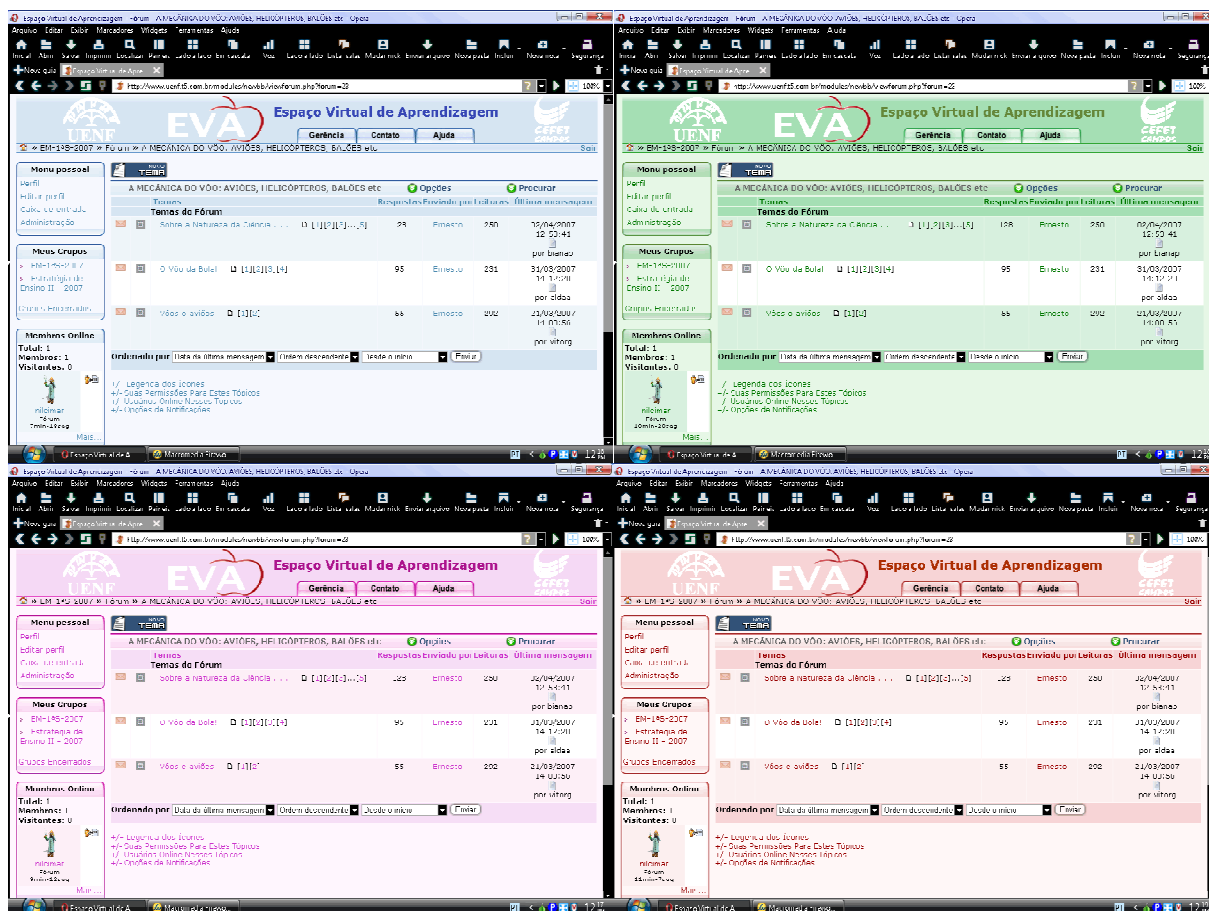
Figura 3.63 – Tela do fórum exibida nas quatro cores disponíveis

O módulo exibe um bloco com os quatro temas listados de forma que o usuário pode escolher e salvar o que preferir. Todos os usuários possuem acesso ao bloco com a opção de alteração de tema.

Kit pedagógico

O Kit Pedagógico, destinatário de materiais didáticos elaborados para auxiliar os alunos na resolução dos Estudos de Caso foi reestruturado conceitualmente, ou seja, o Kit Pedagógico deixou de ser único (“repositório de arquivos” do sistema inicial), para se dividir em quatro seções: Textos, Multimídia, Arte & Entretenimento e Link-Web

As quatro seções mostraram-se subdivisíveis, desta forma é necessário que cada uma delas atenda subseções correlatas, conforme demonstrado no quadro 3.5.

Seção	Subseção
Texto	• Artigos • Divulgação Científica • Educação e Escola • Apostilas • Cadernos e Revistas
Multimídia	• Jogos Online • Painéis • Mapas Conceituais • Applets e Simulações • Software de Modelagem • Outras Ferramentas Cognitivas
Arte & Entretenimento	• Cinema • Exposição e Eventos • Teatro e Outros
Link-Web	• Experimentos • Teorias e Aplicações

Quadro 3.5 – Divisão conceitual do kit pedagógico

O módulo utilizado para implementar o repositório de arquivos [wf-download] foi mantido na versão beta, porém foi utilizado uma cópia do módulo para cada uma das três primeiras categorias, por estarem relacionadas a download de arquivos. Para a quarta categoria, link-Web foi utilizado um módulo de armazenamento e disponibilização de links desenvolvido pela mesma equipe de desenvolvimento do [wf-download] a WF-Project Team. O módulo de link denominado de [wf-links] em conjunto com o [wf-download] formaram a base de desenvolvimento do kit-pedagógico.

As seções Texto, Multimídia e Arte & Entretenimento contaram com cópias do módulo [wf-download], ou seja, optou-se por cada seção ser representada por um módulo independente. Para evitar conflito de dados cada cópia teve seus diretórios, arquivos e tabelas de banco de dados renomeados.

O banco de dados dos módulos [wf-download] prevê o cadastro de categorias e arquivos vinculados a elas. Similar ao bando de dados do [wf-download] o [wf-links] possui banco de dados com categorias e links relacionadas a elas. Estas estruturas mostram-se compatíveis com os requisitos definidos para o módulo de kit pedagógico. Desta forma não foi necessário criação ou exclusão de tabelas ou campos de tabela.

Utilizou-se os módulos de forma a adaptar as necessidades à arquitetura do módulo. Criou-se categorias com os nomes das subseções, por exemplo, no módulo que representa a seção textos, criou-se categorias com os nomes: Artigos, Divulgação Científica, Educação & Escola e Apostilas, que representam as subseções da seção texto. Nelas são armazenados os arquivos e links. Isto configura a ferramenta de bibliotecas, umas das ferramentas da gerência do professor.

No módulo de Estudo de Caso foi criada uma interface em que o professor pode associar cada material do kit pedagógico a um Estudo de Caso, de forma que sempre que o Estudo de Caso for utilizado por algum grupo, ele incorpore automaticamente o kit pedagógico relacionado. Isto configura a ferramenta de gerência do tutor.

Para a exibição dos kits pedagógicos, foi criada uma interface que agrupa os quatro módulos em uma tela disponível a todos os usuários do grupo (figura 3.64).
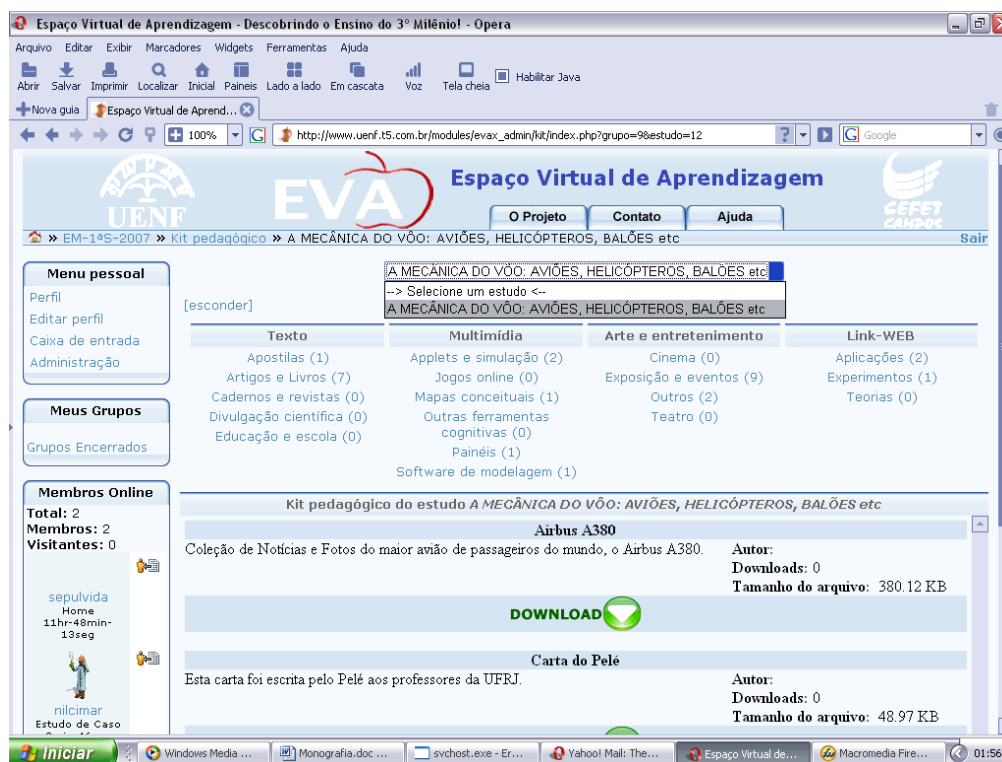
Figura 3.64 – Tela do kit pedagógico exibindo os materiais de uma categoria

Esta tela permite que o usuário selecione um Estudo de Caso dentre os relacionado ao grupo, daí a página busca nos quatro módulos os arquivos relacionados àquele Estudo de Caso e os exibem dentro dos parênteses as quantidades encontradas para cada subseção. Em seguida o usuário seleciona uma subseção e a tela carregará na parte abaixo dos links os arquivos ou links relacionados acompanhados de uma pequena descrição e um ícone de acesso ao arquivo ou o link.

Em virtude da nova interface criada para acessar os recursos do kit-pedagógico, não foi realizada adaptações nas telas originais dos módulos.

Usuários online

Com o objetivo de investir na interação entre os usuários, definiu-se a criação de uma ferramenta que mostre a um usuário, após entrar no ambiente, os logins ativos no mesmo

instante. Isto propicia que um usuário veja os demais usuários logados, e assim possam enviar mensagem privada, marcar conversa no chat e/ou fórum etc.

Após consulta nas comunidades de desenvolvimento relacionadas, verificou-se a existência de alguns módulos com esta funcionalidade, porém o selecionado foi o módulo nacional [m_online] desenvolvido pelo brasileiro Fernando Santos da comunidade de desenvolvimento www.xoopstotal.com.br.

Este módulo exibe em um bloco, visível à todos, uma lista com os usuários logados em um determinado instante, acompanhado de seu tempo de permanência no ambiente e de um ícone que permite o envio de um mensagem privada àquele usuário. Na parte inferior do bloco há um link representado pela palavra “mais...”, que dá acesso a uma tela mais detalhada dos usuários com login ativo e permite a visualização do perfil do usuário.

O bloco com uma lista de usuários e a tela detalhada é apresentado na figura 3.65.
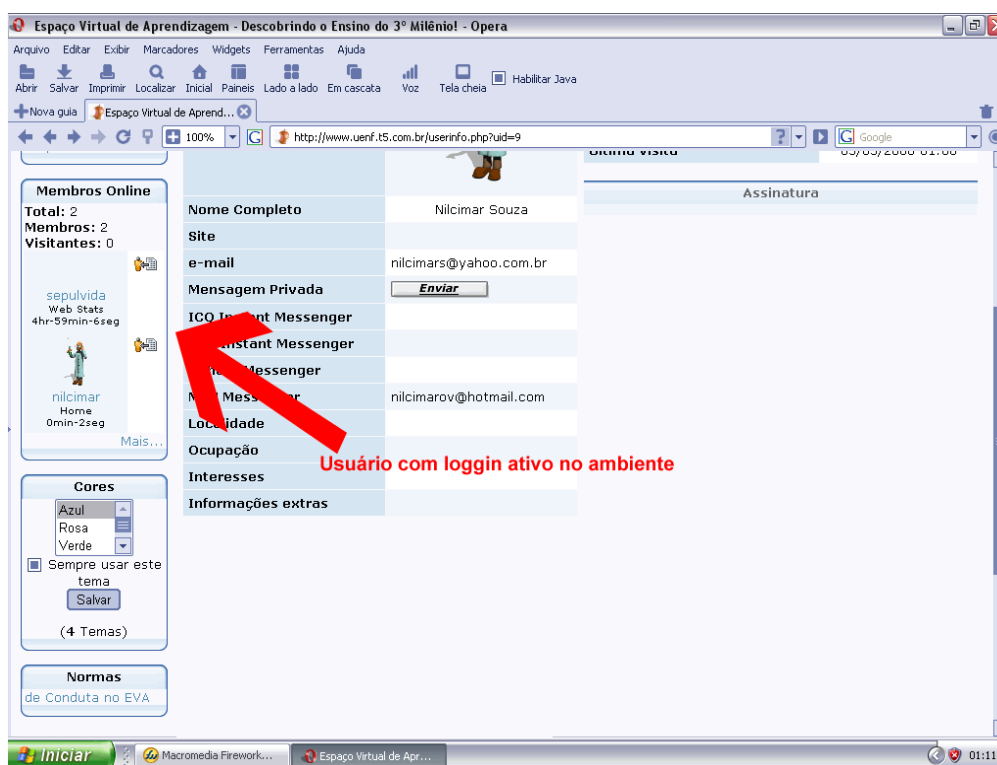
Figura 3.65 – Tela de perfil com destaque ao bloco Membros Online

Na tela destacam-se os usuários com login ativo em um determinado estante.

Estatísticas

A procura por módulos que implementassem as funcionalidades definidas para o módulo de estatísticas, revelou a existência de poucos e limitados módulos com estas características. O que mais se encaixou nas necessidades do sistema foi o módulo [istats]. No

entanto ele necessitou de ampla remodelagem, como descrito em Sepulvida (2008).

3.3.5 – Utilização no Ensino Médio

Para utilização junto a um grupo de estudantes do Ensino Médio contou-se com 24 aulas presenciais distribuídas em dois dias da semana durante um bimestre letivo. Os alunos trabalharam no Laboratório de Física e tinham o EVA como apoio às atividades pedagógicas. Nesse bimestre os estudantes foram cadastrados no EVA durante a oficina (2 aulas iniciais) de familiarização com o ambiente virtual de aprendizagem. Realizou-se o cadastrado de 35 estudantes no EVA, dos quais 28 completaram as atividades e cumpriram todas as tarefas, em sala de aula e no EVA.

Como estratégia de ensino utilizou-se os Estudos de Caso, tendo como tema abordado neste trabalho "A mecânica do vôo de aviões", relacionado ao ano comemorativo do centenário do vôo do 14 BIS. Os estudantes realizaram experiências, fizeram leituras e discussões, construíram mapas conceituais e utilizaram as ferramentas de interação e comunicação do EVA.

Durante a experiência didática, o EVA foi utilizado de forma integrada ao ensino presencial. No laboratório de Física trabalhou-se com atividades experimentais voltadas ao esclarecimento dos conteúdos disciplinares da Física que fazem parte do currículo dessa modalidade pedagógica.

Ações experimentais foram planejadas e executadas em sintonia com o trabalho no ambiente de aprendizagem.

Com o objetivo de avaliar os meios utilizados pelos estudantes para acessar a Internet, a forma de uso e o contato com textos relacionados à ciência, elaborou-se um questionário para que os alunos respondessem no início da experiência didática. Ao término do trabalho, outro questionário foi utilizado com a finalidade de obter a avaliação dos estudantes sobre sua aprendizagem e de que forma o EVA favoreceu essa aprendizagem.

Próximo ao final do ano letivo realizou-se uma entrevista semi-estruturada com 10 estudantes, adotando-se como critério a disponibilidade dos mesmos em permanecerem na escola após o horário da aula. Com as entrevistas espera-se compreender detalhes das participações, pois o caráter individual da mesma, o face a face, favorece o traçar paralelos, identificar desníveis e desvelar quadros humanísticos que estão sempre presentes na pesquisa em educação, e que, conforme Lüdke e André (2001) não podem ser descartados na fase de análise, já que influenciam os resultados.

Capítulo IV

RESULTADOS E ANÁLISE

Neste capítulo estão destacados os resultados da utilização do sistema baseando-se no desenvolvimento da versão beta. É tecida uma análise desses resultados a luz de avaliações de caráter humanísticos e técnicos da interface desenvolvida. O caráter humanístico destaca as interações com os usuários e o caráter técnico os aspectos relacionados à utilização do sistema e seu funcionamento.

Os resultados de caráter humanístico foram obtidos através de questionários e entrevistas semi-estruturadas. Enquanto os resultados técnicos surgiram das observações do sistema em funcionamento e da documentação produzida.

4.1 – Análise dos Questionários

Foram dois os questionários respondidos pelos estudantes do Ensino Médio, o primeiro, no início do ano letivo com o objetivo de levantar o perfil do grupo e o segundo, no final do curso, visando uma avaliação do sistema didático EVA.

O Perfil do Grupo

O questionário aplicado no início do curso – Anexo 2, teve como objetivo ajudar na construção de um perfil do grupo de estudantes.

O grupo apresenta idade média de 16 anos.

Quando perguntados sobre como pretendiam utilizar o computador durante o curso, dos 35 que iniciaram o estudo, 23 apontaram à residência e 12 apontaram outros locais. Dos 12 estudantes que informaram não acessar da residência, 10 citaram o laboratório de Informática da escola como local de acesso.

Outros locais citados foram lan-house e casa de amigo, parente e namorado (figura 4.70). De qualquer forma o laboratório da escola é colocado todo tempo à disposição dos estudantes de 08h00min as 22h00min, o que facilita o acesso, que pode ser feito a qualquer momento de segunda à sexta-feira. Destaca-se que dos 35 estudantes que iniciaram a disciplina, sete não completaram o ano letivo por diferentes razões.
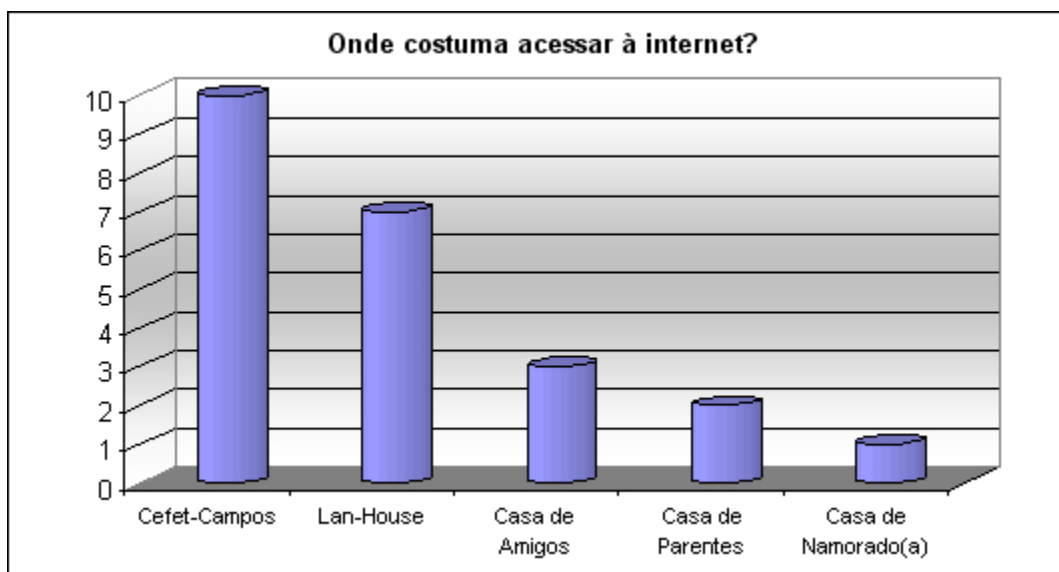
Figura 4.1 – Locais alternativos à residência para acessar o EVA

Dos 23 estudantes que informaram utilizar a residência como principal local de acesso, 16 possuem Internet discada e 7 banda larga. Os demais, 10 não possuem computador na residência e 2 possuem computador, mas sem Internet (figura 4.71).
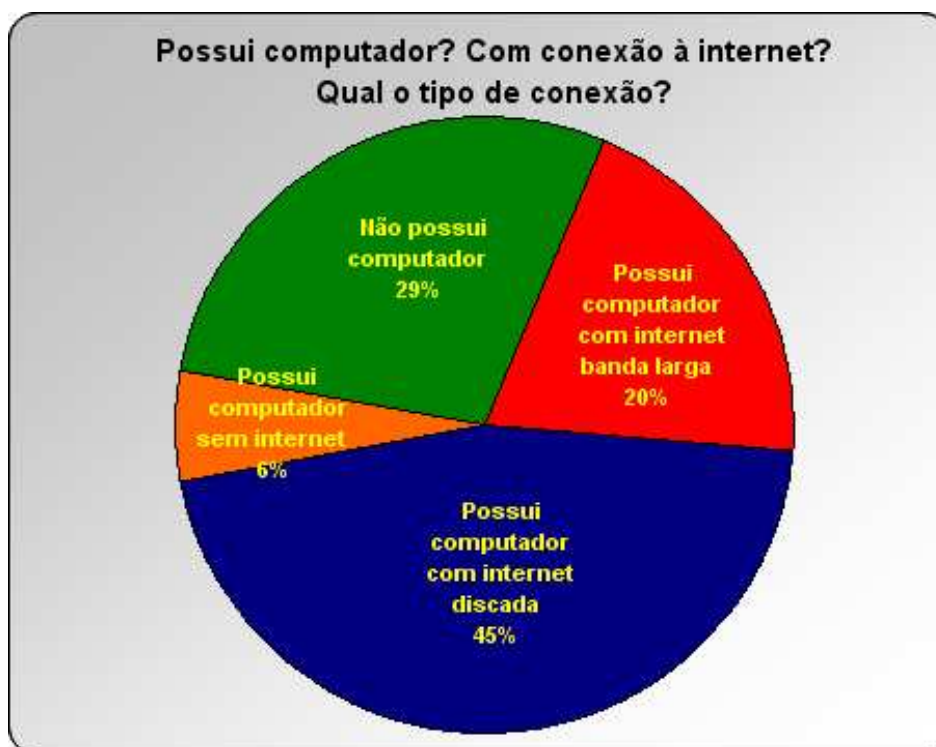
Figura 4.2 – Representação do uso doméstico de computadores pelos estudantes

Inquiridos sobre o número de leituras (livros) não obrigatórias realizadas no ano anterior, obteve-se a média de 0,36, com 23 alunos revelando que não leram livro algum. As leituras executadas pelos demais foram romances, policiais e histórias em quadrinhos. Nenhum estudante fez leituras sobre assuntos relativos à Ciência.

A falta de hábito com leituras por parte dos alunos fez com que o professor disponibilizasse no kit pedagógico do EVA duas leituras sobre aspectos históricos da evolução das idéias da Ciência, (Guerra *et. al*, 1999).

Perguntados sobre suas atividades extra-classe, 5 alunos responderam que só teriam compromisso com a escola, os demais revelaram possuir outras ocupações, sendo as mais freqüentes: atividades esportivas (treinos de futebol, academia ginástica, natação e outras menos cotadas) – 34 estudantes, curso de línguas (inglês) – 21 estudantes, atividades religiosas regulares – 12 estudantes, alguns sobrepunham duas e até três destas atividades.

Sondados sobre o interesse em informática/computadores: 2 alunos revelaram não ter interesse, 33 disseram que sim. Destes, 32 têm contas em orkut e e-mail pessoal, 26 possuem fotolog. Todos os 35 alunos revelaram já ter feito pesquisas utilizando a Internet.

Este era o quadro previsto, pois atualmente não se deve ignorar que os jovens usam computadores e a Internet com destreza, porém, nem sempre de forma organizada e correta.

O Posicionamento dos Estudantes em relação ao Sistema Didático EVA

Ao término da experiência didática, aplicou-se um questionário (Anexo 3) com o objetivo de identificar as impressões dos estudantes sobre o curso e sobre a estratégia de Estudos de Caso adotada. O questionário foi respondido por 28 estudantes da seguinte forma:

1) todos aprovaram a experiência didática com a utilização do EVA;

2) 26 alunos gostariam de continuar utilizando o sistema didático EVA no ano letivo seguinte, 2 alunos não;

3) 27 alunos acreditam que o sistema didático EVA ajudou na aprendizagem de Física, 1 aluno considerou indiferente;

4) todos consideraram que o estudo ficou mais agradável;

5) 27 alunos gostaram mais de trabalhar com a estratégia de estudos de caso quando se utilizou o Espaço Virtual de Aprendizagem, 1 aluno considerou indiferente.

Neste questionário chamou atenção à aprovação da metodologia, praticamente não houve queixas. Esta aprovação revela o envolvimento dos estudantes com a metodologia e se traduz na capacidade do sistema EVA não impor barreiras tecnológicas à sua utilização. Assim os estudantes puderam interagir com o sistema de forma facilitada e sem prejuízos.

4.2 – Entrevistas

As entrevistas (Anexo 4) foram programadas para complementar o quadro da participação dos estudantes no estudo, fundamentalmente no tocante a entender a forma como se percebem atuando em uma metodologia diferenciada das mais tradicionais: Qual foi a impressão que os estudantes tiveram do Curso, da metodologia de ABC, do sistema didático EVA? Gostariam de continuar estudando de acordo com o modelo de Estudos de Caso e fazendo uso do ambiente virtual de aprendizagem? Por que?

A partir do trabalho de Reis (2008) verificou-se elementos das falas dos alunos que podem nos ajudar a entender as formas como utilizaram e aceitaram o EVA.

No quadro 4.1 apresenta-se a fala do estudante A, que utilizou o sistema durante o bimestre letivo em que se fez uso do EVA.

Amei fazer a tarefa da casa da minha avó em Belo Horizonte, pude viajar sem faltar, também gostei de entrevistar meus tios sobre o futebol e a bola, além de dizer para eles coisas que eles não sabiam, só acho que os prazos são muito curtos.

Quadro 4.1 – Depoimento do estudante A

Na fala deste estudante compreende-se que a forma de novidade do estudo foi boa para motivar, flexibilizar a distância (BH é distante mais ou menos 800 km), mas principalmente para gerar interesse em aprender.

Outro aspecto a destacar em complemento ao nosso interesse em avaliar sob que condições os estudantes aprenderam, pode ser percebido na fala de um outro estudante, B:

Assim, eu achei muito interessante, mais não para um estudo avaliativo. Por que, nem sempre agente pode estar na internet, acessando a internet, podendo estar estudando outras coisas né, que agente nem tem todo tempo para estar na internet, para ta lá no fórum pesquisando e tudo mais. O trabalho é muito duro, mais que nas aulas geral, onde todo mundo fica só esperando a prova, aí estuda e pronto.

Quadro 4.2 – Depoimento do estudante B

Pelo depoimento do estudante B, pode-se avaliar a seriedade dos estudos e o nível considerável das cobranças e este fato é destacado, no caso, como uma situação difícil de ser vivenciada. Para este estudante pesquisar foi difícil e o tempo não foi bem administrado

Quando expôs sua satisfação em utilizar o sistema didático EVA, levando seu estudo à esfera familiar o estudante C faz a ressalva sobre o que lhe disse outro professor:

Achei muito bom, foi massa, ajuda muito. Em casa mesmo agente pode fazer os exercícios, as tarefas, na hora em que agente quiser, me ajudou muito nas notas também. Meus irmãos gostaram muito e minha mãe também. Ela até quis usar, mas aí não pode, inclusive eu mesmo não deixei. Porém fica uma dúvida, como disse a professora de Matemática, e no vestibular, não vai dá, tem que ensinar mesmo.

Quadro 4.3 – Depoimento do estudante C

Na fala do estudante C entende-se o posicionamento contrário às inovações tecnológicas. Uma fala mais explícita pertence ao estudante D:

...agente sempre está na internet fazendo alguma coisa, pesquisando , conversando e é bom que agente estuda por lá e aprende assim , aprende por um meio que agente esta sempre em contato , pela internet, bem legal. Não é chato e desinteressante como quando você lê, pega a fórmula e aplica sem saber o que está fazendo, pois nem o professor se preocupa com isso.

Quadro 4.4 – Depoimento do estudante D

Pode-se inferir das falas dos estudantes, que relacionam-se com suas preocupações em relação ao encaminhamento das aulas, quando é adotado o EVA. Uma avaliação sobre tais ponderações dos estudantes não faz parte do escopo deste trabalho.

Outra conclusão que se depreende das falas é a flexibilidade de utilização do sistema EVA. Como todo o sistema informático voltado à Web, o sistema EVA é criado com a perspectiva de permitir que seja acessando de qualquer computador ligado a Internet e em qualquer lugar.

Obteve-se referências, através das entrevistas, sobre a utilização do ambiente na escola, na residência da família e em residência de familiares localizada em outro estado, mas não foi citada por nenhum entrevistado a indisponibilidade do serviço, ou seja, o usuário não conseguir acessar o EVA. Isso demonstra a estabilidade e confiabilidade do sistema.

4.3 – O sistema

A avaliação do sistema foi conduzida através da interação dos usuários com o sistema. A seguir é abordado o desempenho dos módulos e do sistema durante a utilização, além dos conteúdos produzidos. Todas as telas do sistema são apresentadas no anexo eletrônico 2.

Principal

O módulo principal [system] foi responsável, fundamentalmente, pela abertura dos grupos e atribuição das permissões de acesso diferenciadas aos atores do sistema. Neste

contexto não foi percebido falhas. Os usuários foram devidamente associados aos grupos como aluno ou professor e tiveram suas permissões de acesso reconhecidas automaticamente. Ou seja o sistema exibia as informações pertinentes àquele grupo na visão do professor ou aluno. Não houve notificação de problemas e bugs na disponibilização do conteúdo e permissões de acesso.

Estudos de Caso

O módulo de estudos de caso [evax_admin] foi utilizado para o cadastro dos Estudos de Caso e associação ao grupo, cadastro de materiais no kit pedagógico e associação ao grupo, respostas aos passos e correção dos passos. Trabalhou-se um Estudo de Caso neste grupo, gerando 77 soluções ao primeiro passo do Estudo de Caso, 50 soluções para o segundo passo e 29 ao terceiro.

O Número maior de soluções em relação ao número de alunos é devido a solicitação de correção nas soluções feitas pelo professor. Não foram reportados erros no processo de informar ao aluno que a sua solução não foi aprovada e permitir que incluísse nova solução. Não foram identificados erros também, nas visões do professor, ou seja, em ler os as soluções dos alunos e avaliar.

Fórum

O fórum foi utilizado como principal ferramenta de comunicação no estudo. Foram abertos três temas relacionados ao fórum do estudo de caso, produzindo um total de 293 mensagens, que foram lidas 791 vezes ao longo do bimestre. O módulo de fórum utilizado, o [newbb], permitiu não contabilizar as leituras efetuadas pelos administradores, desta forma os valores expressam as leituras realizadas por alunos e professores.

O módulo se mostrou eficaz e motivador, visto os números de mensagem e leituras apresentados, revelando assim o grande interesse dos estudantes em participar. Um estudante demonstra este interesse no trecho de uma mensagem enviada por ele no fórum: "*Eu estou até gostando bastante do fórum. Pesquisei e descobri bastante coisa sobre o vôo da bola. Aí vai minhas conclusões...*".

Avisos

O módulo de avisos [eNoticias], foi utilizado pelos professores para o envio de 8 avisos ao longo do estudo. Os avisos consistiam em informar roteiros para execução dos passo

e notas bimestrais e de recuperação. Os avisos foram lidos 316 vezes pelos alunos. Este módulo não contabiliza as leituras efetuadas pelos administradores.

As modificações realizadas no módulo [eNoticias] para adequar-se aos requisitos de um módulo de avisos, apresentaram-se satisfatórias. Não foram detectados erros no banco de dados e nas interfaces, principais pontos de modificações.

Tarefas

O módulo de tarefas [eNoticias], foi utilizado pelos professores para a publicação de duas tarefas, ambas relacionadas ao Estudo de Caso. As tarefas foram lidas 396 vezes pelos alunos, já que o módulo não contabiliza as leituras efetuadas pelos administradores.

As modificações realizadas no módulo [eNoticias] seguiram a mesma linha do módulo de avisos, assim apresentou de forma igualmente satisfatórias, não sendo detectados erros de banco de dados e de interfaces, principais pontos de modificações.

Kit pedagógico

Os módulos utilizados para compor o kit pedagógico [wf-download] e [wf-links] foram responsáveis pelo cadastro de 8 materiais na categoria de textos, 5 na categoria multimídia, 11 na categoria arte e entretenimento e 3 na categoria link-Web. Não foi possível a obtenção da quantidade de download de materiais e acesso aos link-Web efetuados pelos alunos.

Apesar de não possibilitar a extração da quantidade de vezes que o aluno acessou determinado arquivo ou link-Web, os módulos que compõem o kit pedagógico mostraram-se funcionais, estáveis e atenderam adequadamente a necessidade de disponibilizar materiais didáticos aos grupos de estudo.

Chat

O módulo de chat [khat] não foi utilizado de maneira formal durante a experiência, no entanto, por estar disponibilizado à acesso, alguns alunos enviaram mensagens, sem muito conteúdo e com o objetivo de iniciar conversa com algum colega online. Foram enviadas 27 mensagens. Isto mostra a relação dos estudantes com esta ferramenta e a vontade de trabalhar com tais recursos.

Apesar de não haver preparação para uma utilização mais direcionada, percebeu-se que o chat é potencialmente uma ferramenta importante no contexto didático-pedagógico e flexível no contexto do sistema.

Ajuda

O módulo destinado a publicação de tópicos de ajuda [smartyFAQ] foi utilizado para a publicação de um tópico, o qual contém um manual destinado aos alunos, ilustrando e comentando todas as telas acessíveis por eles. O manual encontra-se em duas versões, uma para usuários com Internet banda larga e outra para usuários, em arquivos particionados, para usuários com Internet discada. Este tópico foi lido 59 vezes pelos alunos, já que o módulo não contabiliza acessos de professores.

A ajuda do ambiente atualmente abrange apenas as visões dos alunos, um manual para o professor ainda não foi disponibilizado. A seção ajuda em todo software é um espaço que poucos usuários recorrem, acabam preferindo elucidar suas dúvidas na Internet ou com amigos, no entanto no EVA pode-se afirmar que a seção foi relativamente bem utilizada e se mostra com um amplo potencial de administração de tópicos de ajuda.

Cronograma

O módulo de cronograma não foi utilizado pelos estudantes do Ensino Médio.

Contato

O formulário para contatos desenvolvido no módulo [liaise] permitiu que os usuários entrassem em contato com os administradores para solucionar alguma dúvida não esclarecida pela ajuda. Durante a utilização do EVA os administradores receberam em seus e-mails pessoais 19 mensagens, das quais 13 pertenciam a um aluno que erroneamente utilizou-se desta ferramenta para encaminhar soluções às tarefas propostas. As demais mensagens relacionavam-se a dúvidas de como abordar as soluções dos passos aos Estudos de Caso. Tais mensagens foram encaminhadas aos professores, devido seus conteúdos.

Nota-se a inexistência de mensagens relacionadas à utilização do sistema, revelando que qualquer dúvida surgida durante a utilização foi contornada através de tentativas e erros, ajuda de amigos ou do manual do aluno disponibilizado na ajuda.

É possível destacar a eficiência da ferramenta, todos os administradores receberam as 19 mensagens em seus e-mails e não houve relatos de estudantes, que tenham enviado uma mensagem e não obtido resposta.

Usuários online

O módulo que possibilita a exibição dos usuários com login ativo não possui formas de contabilizarmos quantas vezes foi utilizado pelos usuários na verificação dos usuários online.

Os resultados deste módulo baseiam-se em observações realizadas durante as oficinas, onde se percebeu que o recurso foi bastante utilizado pelos usuários para enviar mensagens particulares aos demais usuários online.

Desta forma, o módulo cumpriu de forma satisfatória sua funcionalidade de exibir os usuários online e motivou os estudantes a interagir através da ferramenta de mensagens particular.

Texto Estático

O módulo para exibição de textos estáticos executa uma das atividades mais simples na arquitetura do sistema EVA, que é a exibição de telas escritas em HTML. As telas desenvolvidas e exibidas no módulo mantiveram sua formatação original, desenhada para apresentação. Desta forma o módulo [xt_conteudo] se mostrou eficaz para exercer esta função no sistema.

Personalização da Interface

O módulo de personalização de interface [xt_temas] foi o módulo mais utilizado no EVA, fora do contexto didático-pedagógico do ambiente.

Os usuários utilizaram de forma acentuada esta ferramenta na intenção de adequar sua interface de acordo com suas preferências. Não é possível determinar quantas vezes cada usuário trocou as cores de sua interface, no entanto é possível determinar que após o encerramento do grupo 12 estudantes utilizavam o EVA com o tema cores azul, 13 com o tema rosa, 8 o tema verde e 2 o tema vermelho.

Considerando-se que o tema padrão para todos os usuários é o azul, vê-se a alta interação dos estudantes com o módulo, que revela a necessidade dos usuários poderem adequar a foram de apresentação às suas preferências e não ficarem presos a um modelo de apresentação padrão e obrigatório.

É possível percebermos a adequação do módulo a esta necessidade, já que não houve relatos de erros durante a substituição das cores de apresentação da interface.

Backup

Os módulos utilizados para backup do banco de dados foram fundamentais para os administradores do sistema. Durante a utilização, os administradores utilizaram os módulos para efetuar constantes e periódicos backups do banco de dados, que foram utilizados para testes nos computadores de desenvolvimento. Não foi necessário utilizar backups para restaurar dados corrompidos no banco, já que o sistema não apresentou problemas dessa natureza.

Todos os backups foram classificados como representantes fies a estrutura do banco de dados, revelando que os módulos utilizados neste processo são bem estruturados e confiáveis para a manutenção de políticas de segurança de dados.

Estatísticas e Relatórios

As ferramentas de estatística e relatórios foram fundamentais na pesquisa, pois possibilitou a obtenção de resultados necessários no processo de avaliação do sistema, dos estudantes e numa fase posterior fornecer material para auxiliar a publicação e divulgação acadêmica.

Os resultados detalhados destas ferramentas são expostos em Sepulvida (2008).

Capítulo IV

CONSIDERAÇÕES FINAIS

Sistemas de Gerenciamento de Conteúdo ajudam a superar potenciais barreiras à comunicação em ambientes virtuais de aprendizagem, reduzindo o custo da criação e manutenção de conteúdo nas aplicações de software, respondendo-se a principal questão tratada nesta monografia. A participação no projeto de pesquisa iniciou com este objetivo, confirmando após 2 anos de trabalhos de desenvolvimento, que o XOOPS se mostra uma boa ferramenta para desenvolvimento de ambientes virtuais de aprendizagem.

O estágio atual de desenvolvimento do EVA não explorou todos os recursos do XOOPS. Algumas potencialidades do SGC como um editor de texto WYSIWYG não foi incorporado à estrutura do sistema. Os recursos foram inseridos à medida que se fazia necessário. Utiliza-se no sistema 16 módulos distintos, ao passo que no site oficial do XOOPS, existem 405 módulos cadastrados. Isto demonstra o vasto campo de trabalho, melhorias e pesquisa que podem ser desenvolvidos sobre o sistema.

Grande parte dos 405 módulos contidos na comunidade oficial do XOOPS é desenvolvida por empresas, porém possuem alguma licença relacionada a General Public License – GPL, que torna o módulo livre para uso e modificação. Isto revela como os recursos destinados aos SGC, inicialmente empresariais, encaminham-se para uma abordagem de código livre.

A arquitetura modular utilizada para a codificação se mostrou adequada, pois permitiu o gerenciamento de forma pontual e segmentada, por exemplo, as permissões, inicialmente planejadas para serem atribuídas apenas pelo módulo principal, foram subdivididas entre todos os módulos, que necessitam exibir conteúdo em mais de uma visão.

As teorias referentes à engenharia de software foram perfeitamente aplicadas ao desenvolvimento do Espaço Virtual de Aprendizagem, viabilizando a aplicação e utilização do sistema gerenciador de conteúdo XOOPS de forma correta. Uma pesquisa realizada pela empresa norte-americana de consultoria em projetos de tecnologia da informação *The Standish Group International avilia*, no ano de 2000, mostra que apenas 28% dos projetos são bem sucedidos, ou seja, são finalizados no tempo e orçamento estimados e com todas as funcionalidades acordadas implementadas. Dos demais projetos, 49% são finalizados, porém com prazos e custos ultrapassados, e 23% nem chegam a ser concluídos.

A utilização do XOOPS no desenvolvimento do EVA, flexibilizado pelos padrões de engenharia de software, permitiram que o sistema fosse entregue dentro dos prazos

estipulados, além de apresentar resultados com o pré-teste e posteriormente com a utilização no Ensino Médio.

As atividades de documentação ao longo de todo o desenvolvimento permitiram compreender o sistema à medida que as modificações eram necessárias. Espera-se que toda a documentação produzida seja suficiente para que novos mantenedores compreendam e modifiquem o sistema para aplicação de acordo com novas realidades de uso.

A utilização foi essencial para a compreensão do sistema através da percepção dos usuários, os questionários foram importantes para refletir as expectativas e avaliações dos usuários. Já as entrevistas permitiram desvelar quadros humanísticos não explicitados pelos estudantes nos questionários, mas que influenciaram os resultados.

Quanto a participação no projeto, o desconhecimento inicial da área de Informática Educativa, transformou-se em um vislumbre de novas possibilidades para a formação como Tecnólogos em Desenvolvimento de Software.

Após a participação pretende-se dar continuidade a esta formação, participando do trabalho de campo em salas de aula, pois como licenciando em Química em fase de conclusão de curso é possível reconhecer a existência de caminhos na área da educação que precisam ser trabalhados por profissionais com múltiplas formações. É isso que se tem em mente, pois a InfoEduc se mostra numa área limítrofe do conhecimento entre inúmeras que se relacionam à educação.

É possível que devido ao crescimento mundial da InfoEduc, o CEFET-Campos possa incluir brevemente no currículo do Tecnólogo em Informática, disciplinas de Informática Educativa, que nesse momento é objeto de interesse também na instituição, por parte de diversos professores que buscam formação de doutoramento nessa área.

ADIERS, D. R. **Gerenciamento de Conteúdo na Web Usando PLONE: Aplicação ao Portal da Informática da UFSM**. UFSM - Universidade Federal de Santa Maria, RS. 2007. Projeto de Final de Curso de Graduação em Ciência da Computação.

ALVES, P., PIRES, J. A. **A usabilidade em software educativo: princípios e técnicas**. In: IE 2002 - 6 Congresso Iberoamericano, 4º Simpósio Internacional de Informática Edicativa, 7º Taller Internacional de Softwrae Educativo, 2002, Vigo. IE 2002 - 6 Congresso Iberoamericano, 4º Simpósio Internacional de Informática Edicativa, 7º Taller Internacional de Softwrae Educativo. Vigo, Espanha : Servicio de Publicacións da universidade de Vigo, 2002.

ARAÚJO, R. S. (2005). **Formação continuada a distância de professores de física: o desenvolvimento do conhecimento profissional**. Dissertação (Mestrado em tecnologia educacional) – NTE, UFRJ, Rio de Janeiro, 172p.

ATWAL, S. **Building Websites with XOOPS: a step-by-step tutorial**. Birminghan: Packt Publishing, 2006.

AZEVEDO, E. (2003). **Desenvolvimento de um Ambiente Virtual para o Ensino de Ciências.** Dissertação de Mestrado – Universidade Estadual do Norte Fluminense Darcy Ribeiro: Programa de Engenharia de Produção, mar: Campos dos Goytacazes, 111p.

BOEHM, B.W. **Verifying and validating software requirements and design specification**. IEEE software Vol 1 pp-94-103, Jan 1984.

BOOCH, G.; RUMBAUGH, J.; JACOBSON, I.; (2000). **UML, Guia do usuário.** Campus, Rio de Janeiro.

CORTES, S. C.; LUCENA, C. J. P. Um framework para construção de sistemas de banco de dados móvel com regras ativas. 2001.

CYSNEIROS, L. M. (2001). **Requisitos Não Funcionais: Da Elicitação ao Modelo Conceitual**. Pontifícia Universidade Católica do Rio de Janeiro, PUC/RJ. (Tese de Doutorado).

DILLENBOURG, P. (2003). **Virtual Learning Environment**. (Acesso maio 2007). Disponível em: <http://tecfa.unige.ch/tecfa/publicat/dil-papers-2/Dil.7.5.18.pdf>.

FREIRE, P. (1970). **Pedagogia do Oprimido**. Ed. Paz e Terra, Rio de Janeiro. 184p.

GEROSA, M. A. (2006). **Desenvolvimento de Groupware Componentizado com Base no Modelo 3C de Colaboração**. Pontifícia Universidade Católica do Rio de Janeiro, PUC/RJ. (Tese de Doutorado).

JONASSEN, D. (1998). Designing Constructivist Learning Environments In C. M. Reigeluth (ED). **Instructional Theories and models**. 2nd Ed. Mahwah. NJ. Lawrence. Erlbaum.

MORAES, M. C. **Informática Educativa no Brasil: Uma história vivida, algumas lições aprendidas**. Revista Brasileira de Informática na Educação, 1997.

PAMPLONA, M. H. et al. Análise de avaliação: contribuições de uma ferramenta de fórum para o ensino de física no nível médio. In: Simpósio Brasileiro de Informática Educativa, XVII., 2006, Brasília. Artigo **Anais do XVII Simpósio Brasileiro de Informática Educativa**.

PARNAS, D. L. **On Criteria to be used in Decomposing Systems into Modules**, CACM, vol. 14, nº 1, abril de 1972, pp.221-227.

PAULA, F. C. e HORA, T. S. **Desenvolvimento de Ferramentas de Informação e Comunicação para Suporte de um Ambiente de Aprendizagem na Web**. CEFET-Campos, 2002. Projeto de Final de Curso de Tecnólogo em Informática.

PEQUENO, M. C., LOUREIRO, R. C., SARMENTO, W. W. F., PAULA, P. S., PEQUENO, H. S. L., SILVA, C. L. O. da . **Desenvolvimento de um Ambiente Virtual de Aprendizagem**. In: IE 2002 - 6 Congresso Iberoamericano, 4º Simpósio Internacional de Informática Edicativa, 7º Taller Internacional de Softwrae Educativo, 2002, Vigo. IE 2002 - 6

Congresso Iberoamericano, 4º Simpósio Internacional de Informática Edicativa, 7º Taller Internacional de Softwrae Educativo. Vigo, Espanha : Servicio de Publicacións da universidade de Vigo, 2002.

PRESSMAN, R. S. (2007). **Engenharia de Software**. Pearson Makron books, São Paulo, 1056p.

REIS, E.M. (2001). **Desenvolvimento e Avaliação de um Ambiente Construtivista de Aprendizagem a Distância na Formação Continuada de Professores de Física do Nível Médio na Internet**. Dissertação de Mestrado, IM/NCE/UFRJ. jun. Rio de Janeiro. 106p.

REIS, E. M. (2008). **Limites e Possibilidades de Utilização de um Espaço Virtual de Aprendizagem no Ensino e na Formação de Professores de Física.** Tese de Doutorado apresentada ao PRPG em Ciências Naturais, UENF, Campos dos Goytacazes, 347p.

REIS, E. M., LINHARES, M. P. (2005b). Convergências Tecnológicas: redesenhando as fronteiras da formação de professores de ciências. In: V Encontro de Pesquisa em Educação em Ciências, Bauru, **Anais do V ENPEC**, CD-ROM.

ROSELLÓ, E. G., DACOSTA, J. G., MANDADO, E., PARDO, V. G. V., PÉRES-SCHOFIELD, J. B. G., COTA, M. P. **Una propuesta para la reutilización de componentes en el proceso de desarrollo de software educativo**. In: IE 2002 - 6 Congresso Iberoamericano, 4º Simpósio Internacional de Informática Edicativa, 7º Taller Internacional de Softwrae Educativo, 2002, Vigo. IE 2002 - 6 Congresso Iberoamericano, 4º Simpósio Internacional de Informática Edicativa, 7º Taller Internacional de Softwrae Educativo. Vigo, Espanha : Servicio de Publicacións da universidade de Vigo, 2002.

SAVERY, J. R. & DUFFY, T. M. (1995). **Problem Based Learning: An Instructional Model and Its Constructivist Framework Educational Technology**. September-October, p. 31-37.

SCHANK, R. C. & CLEARY, C. (1995). **Engines for Education**. Hillsdale, NJ, Lawrence Erlbaum Associates.

SEPULVIDA, L. M. **Desenvolvimento das Ferramentas Relatórios e Estatísticas do Espaço Virtual de**. CEFET-Campos, 2008. *Pré print*, Projeto de Final de Curso de Tecnólogo em Informática..

SILVA, L. F. da. (2006). **Uma Estratégia Orientada a Aspectos para Modelagem de Requisitos**. Pontifícia Universidade Católica do Rio de Janeiro, PUC/RJ. (Tese de Doutorado).

SILVESTRINI, S. (2007). **XOOPS: Creare siti dinamici con facilità.** Manhattan (New York), Stevensword, Inc., 168p.

SOMMERVILLE, I. (2007). **Engenharia de Software**. Pearson Prentice Hall, São Paulo. 548p.

SOUZA, Nilcimar dos Santos; LINHARES, Marília Paixão; SEPULVIDA, Lucas Manhães; REIS, Ernesto Macedo (2007a). **Uma Experiência sobre Avaliação de Conteúdo com Fórum no Ensino de Física**. In: VII Encuentro Internacional Virtual Educa, 2007, São José dos Campos. Anais do VII Encuentro Internacional Virtual Educa.

SOUZA, Nilcimar dos Santos; SEPULVIDA, Lucas Manhães; CABRAL, Éderson Gonçalves Dutra; REIS, Ernesto Macedo (2007b). **Educando para a ciência: engenharia de um ambiente virtual de aprendizagem**. In: 59 Reunião da Sociedade Brasileira para o Progresso da Ciência, 2007, Belém. Anais da 59 Reunião da SBPC.

STRUCHINER, M. REZENDE, F. RICCIARDI, R. M. V. CARVALHO, M. A. P. (1998). **Elementos Fundamentais para o Desenvolvimento de Ambientes Construtivistas de Aprendizagem a Distância.** Tecnologia Educacional – v. 26 (142) jul/set.

TAYLOR, E. S., **An Interim Report on Engineering Design**, Massachusetts Institute of Technology, 1959.

VALENTE, J. A. O uso inteligente do computador na Educação. Pátio – **revista pedagógica**, Porto Alegre, Ano I, No 1, p.18-21, maio/julho 1997.

XAVIER, C.M.S., PORTILHO, C. **Projetando com Qualidade a Tecnologia de Sistemas de Informação**. Livros Técnicos e Científicos Editora, 1995.

ANEXO 1

Tabela comparativa entre os Sistemas Gerenciadores de Conteúdo (SGC) selecionados como possíveis ferramentas de apoio ao desenvolvimento do sistema EVA.

Requisitos de sistema	**PHP-Nuke**	**Xaraya**	**Xoops**
Servidor de Aplicação	mod_php	?	PHP 4 or 5, with XML support.
Banco de dados	MySQL, Postgres, mSQL, Interbase, Sybase	MySQL, MySQLi, PostgreSQL, SQLite	MySQL 4.23.xx or later
Sistema Operacional	Qualquer	Qualquer	Qualquer
Linguagem de Programação	PHP	PHP	PHP 4.1.0 or later
Acesso de Root	Sim	Não	Sim
Acesso à kernel	Sim	Não	Sim
Web Server	Apache, IIS	Any php enabled server	Apache, IIS
Segurança	**PHP Nuke**	**Xaraya**	**Xoops**
Auditoria	Não	Plug-in	Limitado
Captcha	Não	Plug-in	Plug-in
Aprovação de conteúdo	Não	Plug-in	Sim
Verificação de E-mail	Não	Sim	Sim
Permissões Granulares	Limitado	Sim	Sim
Autenticação Kerberos	Não	Não	Não
Autenticação LDAP	Não	Plug-in	Sim
Histórico de login	Não	Não	Plug-in
Autenticação NIS	Não	Não	Não
Autenticação NTLM	Não	Não	Não
Autenticação Pluggable	Não	Sim	Não
Notificação de Problema	Não	Não	Limitado
E-mail interno	Não	Plug-in	Sim
Gerenciamento de Sessão	Não	Sim	Limitado
Autenticação SMB	Não	Não	Não
Compatível com SSL	Não	Sim	Sim
SSL Logins	Não	Sim	Sim
Páginas SSL	Não	Sim	Sim
Controle de versões	Não	Plug-in	Sim
Suporte	**PHP Nuke**	**Xaraya**	**Xoops**
Programa de certificação	Não	Não	Sim
Esqueleto de código	Não	?	Não
Manuais	Não	Limitado	Sim
Suporte	Não	Sim	Sim
Treinamento	Não	Sim	Não
Comunidades de Desenvolvimento	Sim	Sim	Sim
Ajuda online	Não	Sim	Limitado
API plugável	Sim	Sim	Sim
Hosting profissional	Sim	Sim	Sim
Serviços profissionais	Não	Sim	Sim
Fórum Público	Sim	Sim	Sim
Lista e-mail público	Não	Sim	Sim
Teste de Framework	Não	?	Não
Desenvolvimento de terceiros	Sim	Sim	Sim
Conferência de Usuários	Não	Não	Sim
Facilidade de Uso	**PHP Nuke**	**Xaraya**	**Xoops**
Conteúdo Drag-N-Drop	Não	Não	Plug-in
E-mail para discussão	Não	Plug-in	Não

URLs amigáveis	Não	Sim	Plug-in
Redimensionamento de imagens	Não	Não	Plug-in
Macro Language	Não	Sim	Limitado
Upload Massivo	Não	Plug-in	Plug-in
Prototipagem	Não	Sim	Não
Servidor de linguagem de páginas	Sim	Sim	Sim
Site Setup Wizard	?	?	?
Corretor ortográfico	Não	Plug-in	Não
Style Wizard	Não	?	Não
Assinatura	Não	Plug-in	Sim
Template	Não	Sim	Sim
Níveis de interface de usuário	Não	Sim	Sim
Desfazer	Não	Plug-in	Limitado
Editor WYSIWYG	Não	Plug-in	Plug-in
Arquivos ZIP	Não	?	Não
Desempenho	**PHP Nuke**	**Xaraya**	**Xoops**
Cache Avançado	Não	Sim	Não
Replicação de Banco de Dados	Não	Não	Não
Balanceamento de Carga	Não	Sim	Sim
Cache de Página	Não	Sim	Sim
Exportação de Conteúdo estatístico	Não	Não	Não
Gerenciamento	**PHP Nuke**	**Xaraya**	**Xoops**
Gerenciamento de advertências	Sim	Plug-in	Sim
Gerenciamento Asset	Não	Plug-in	Sim
Clipboard	Não	Não	Sim
Agendamento de Conteúdo	Não	Plug-in	Sim
Content Staging	Não	Sim	Limitado
Administração Inline	Não	Não	Sim
Administração Online	Sim	Sim	Sim
Disponibilização de Pacote	Não	Sim	Sim
Sub-sites / Roots	Não	Sim	Plug-in
Temas / Skins	Sim	Sim	Sim
Lixeira	Não	Plug-in	Não
Estatísticas Web	Sim	Plug-in	Plug-in
Gerenciamento de templates / Estilos baseado na Web.	Limitado	Não	Sim
Gerenciamento de traduções baseados na Web	Não	Sim	Sim
Ferramenta Workflow	Não	Plug-in	Não
Interoperabilidade	**PHP Nuke**	**Xaraya**	**Xoops**
Content Syndication (RSS)	Não	Sim	Sim
Suporte FTP	Não	Não	Sim
iCal	?	?	?
Suporte UTF-8	Não	Sim	Limitado
WAI Compliant	Não	Limitado	Não
Suporte WebDAV	Não	Sim	Não
Aceita XHTML	Não	Sim	Não
Flexibilidade	**PHP Nuke**	**Xaraya**	**Xoops**
Suporte CGI-mode	Sim	Sim	Não
Reuso de Conteúdo	Não	Sim	Plug-in
Estender o perfil do usuário	Não	Sim	Não
Localização de Interface	Sim	Sim	Sim
Meta dados	Não	Sim	Plug-in
Conteúdo Multi-linguagem	Não	Sim	Plug-in
Integração de Conteúdo Multi-linguagem	Não	Sim	Plug-in
Disponibilização de Multi-Site	Não	Sim	Não
Reescreve URL	Não	Sim	Sim

Aplicações	PHP Nuke	Xaraya	Xoops
Blog	Sim	Plug-in	Plug-in
Chat	Não	Plug-in	Plug-in
Classificados	Não	Não	Plug-in
Gerenciamento de Contatos	Não	Plug-in	Sim
Data Entry	Não	Plug-in	Plug-in
Relatórios do Banco de Dados	Não	Plug-in	Plug-in
Discussão / Fórum	Sim	Plug-in	Sim
Gerenciamento de Documentos	Não	Plug-in	Plug-in
Calendários de Eventos	Não	Plug-in	Plug-in
Gerenciamento de Eventos	?	?	Plug-in
Relatórios de despesas	Não	Não	Não
Gerenciamento de FAQ	Sim	Plug-in	Sim
Distribuição de Arquivos	Sim	Plug-in	Plug-in
Gráficos e diagramas	Não	Plug-in	Não
Groupware	Não	Plug-in	Não
Livro de visitas	Não	Plug-in	Plug-in
Help Desk / Bug Reporting	Não	Plug-in	Plug-in
HTTP Proxy	Não	Não	Sim
In/Out Board	Não	Não	Não
Job Postings	Não	Plug-in	Plug-in
Gerenciamento de link	Sim	Plug-in	Sim
Formulário de e-mail	Não	Plug-in	Sim
Matrix	Não	?	Não
Página pessoal	Não	Sim	Sim
Newsletter	Não	Plug-in	Plug-in
Galeria de Fotos	Não	Plug-in	Plug-in
Enquetes	Sim	Plug-in	Sim
Gerenciamento de Produtos	Não	Plug-in	Plug-in
Project Tracking	Não	Plug-in	Plug-in
Ferramenta de Busca	Não	Plug-in	Sim
Mapa do Site	Não	Plug-in	Plug-in
Stock Quotes	Não	?	Plug-in
Surveys	Não	Plug-in	Sim
Syndicated Content (RSS)	Sim	Plug-in	Sim
Testes / Quizzes	Não	Não	Plug-in
Time Tracking	Não	Não	Plug-in
Contribuições de usuários	Sim	Plug-in	Sim
Clima	Não	?	Plug-in
Web Services Front End	Não	Sim	Plug-in
Wiki	Não	Limitado	Plug-in
Comércio	**PHP Nuke**	**Xaraya**	**Xoops**
Affiliate Tracking	Não	Plug-in	Sim
Gerenciamento de inventário	Não	Não	Plug-in
Pagamento de itens plugáveis	Não	Plug-in	Plug-in
Pluggable Shipping	Não	Não	Plug-in
Pluggable Tax	Não	Não	Plug-in
Ponto de venda	Não	Plug-in	Não
Carrinho de compras	Não	Limitado	Plug-in
Subscriptions	Não	Limitado	Plug-in
Lista de desejos	Não	Não	Plug-in

ANEXO 2

QUESTIONÁRIO: Sobre as aulas de Física no Ensino Médio e os interesses Alunos

Colégio: ____ data: ___/___/____

1. Sobre as aulas de Física e os estudos que fazem.
1.a) Série em que estuda: ( ) 1ª. ( ) 2ª. ( ) 3ª.

1.b) Você gostaria que as suas aulas de Física fossem diferentes? ( ) Sim ( ) Não

1.c) No momento, como você dispõe seu tempo para estudar Física? Assinale "EM" no seu turno de estudo e coloque um "X" nos outros horários ocupados (compromisso) com qualquer tipo de atividade, escolar ou não. Deixe em branco os horários que você disponibiliza para estudar além do horário escolar.

Horário	Domingo	Segunda	Terça	Quarta	Quinta	Sexta	Sábado
Manhã							
Tarde							
Noite							

2. Sobre as condições pessoais e, de estudo na escola e fora dela.

2.a) Quantos anos você tem? ____________anos completos

2.b) Você utiliza computador regularmente? ( ) Sim ( ) Não

2.c) Possui computador em casa?
( ) Não
Em caso negativo, seu acesso a computador se dá, preferencialmente em:
( ) casa/residência
( ) Lan House
( ) no trabalho
( ) outro: ____

( ) Sim, { ....sem acesso à Internet ( )
.......com acesso à Internet ( ) { ...banda larga ( )
...conexão discada ( ) }

2.d) Possui Instant Messenger (MSN, ICQ, GoogleTalk etc)? ( ) sim ( ) não

2.e) Possui Blog ou Fotolog? ( ) sim ( ) não

2.f) Possui conta no Orkut? ( ) sim ( ) não

2.g) Já utilizou a Internet para realizar alguma pesquisa? ( ) sim ( ) não

2.h) Você tem aulas em laboratório de Física normalmente? (   ) Sim   (   ) Não

2.i) Você utilizava regularmente seu livro de Ciências/Física no ano anterior?

(   ) Sim

(   ) Não

(   ) Não tinha um livro de Física adotado.

2.j) Quantos livros relacionados às Ciências Naturais você leu nos últimos 12 meses?

(   ) nenhum

(   ) um

(   ) dois

(   ) três

(   ) quatro

(   ) mais de quatro

Caso você tenha feito alguma (s) leitura (s), identifique-a (as): ________________________

____

____

____.

3) Espaço livre para você falar o que quiser sobre suas aulas e o ensino de Física.

____

____

____

____

____

ANEXO 3

QUESTIONÁRIO

AVALIAÇÃO DO ESPAÇO VIRTUAL DE APRENDIZAGEM

CLASSE – Estudantes do Ensino Médio / CEFET-Campos Em: _____/____/______

Nome:____ Turma:__________

1. Você aprovou a experiência didática de utilização do EVA?
( ) NÃO ( ) SIM

2. Você gostaria de continuar utilizando este sistema em apoio às aulas no próximo ano letivo?
( ) NÃO ( ) SIM

3. Você considera que o EVA auxilia na aprendizagem de Física?
( ) NÃO ( ) SIM ( ) INDIFERENTE

4. O estudo ficou mais agradável?
( ) NÃO ( ) SIM

5. Você gostou de trabalhar mais com os Estudos de Caso quando utilizou-se o EVA?
( ) NÃO ( ) SIM ( ) INDIFERENTE

6. O que você destacaria no EVA, sob os seguintes aspectos:
POSITIVO:____
____
____
____.
NEGATIVO:____
____
____
____.

7. Espaço livre destinado a sugestões sobre o sistema e sua utilização.
____
____
____
____
____
____

Obrigado por sua participação no desenvolvimento do EVA

ANEXO 4

Protocolo de Entrevista

Protocolo de Entrevistas

Questão 1: Qual a importância do tema "A Mecânica do Vôo de Aviões"?

Questão 2: O Que você achou do EVA – Espaço Virtual de Aprendizagem?